

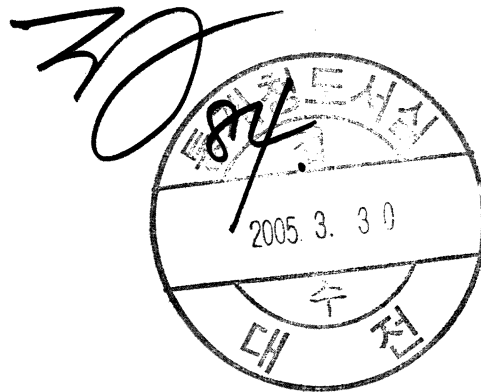
資料處理教育用
84-16

3/10, 12/1  
78746

# 人口動態申告調查

(資料處理 및 터미날 利用方法)

1984. 5.



340829

經濟企劃院 調查統計局

이 冊은 人口動態申告資料處理(出生, 死亡, 婚姻, 離婚) 作業時 必要한 結果表算出, DATA 修正 등을 터미날에서 누구나 손쉽게 應用할 수 있도록 說明이 되어 있습니다.

作業處理시 疑問點이 있으면 資料處理課로 問議하십시오.

資料處理課長    최    돈    철

問議處( 305 )    김    형    근

김    설    희

최    순    애

# 目 次

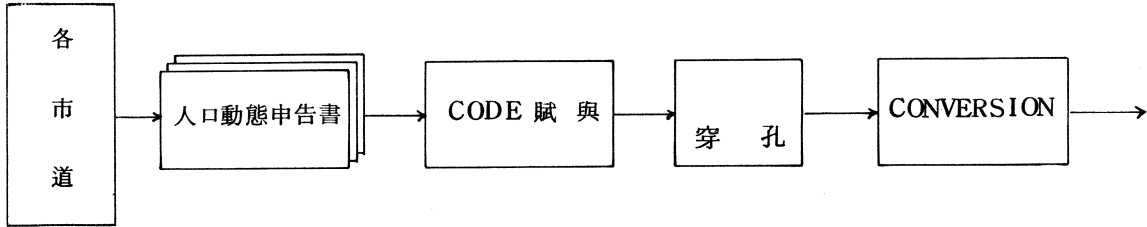
<b>1. SYSTEM FLOWCHART</b> .....	5
1) 當年分 .....	7
2) 10年分 .....	8
3) 死因分類 .....	10
<b>2. 入力 DESIGN LAYOUT</b> .....	11
1) 當年分 .....	13
2) 10年分 .....	14
<b>3. SOURCE PROGRAM LIST</b> .....	15
凡. 出生 .....	17
1) SVBWTT14 : RIGHT JUSTIFY 및 BLOCK化 .....	17
2) SVBWCR 14 : TAPE TO DISK .....	19
3) SVBWET14 : EDIT (出生, 死亡, 婚姻, 離婚, 共同使用) .....	21
4) SVBWPB14 : UP - DATE ( 1次 ) 出生, 死亡, 婚姻, 離婚, 共同使用 ...	48
SVBWAC 14 : UP - DATE ( 2次 ) 出生, 死亡, 婚姻, 離婚, 共同使用...	51
5) SVBWBK14 : DISK TO TAPE 出生, 死亡, 婚姻, 離婚, 共同使用 ...	52
6) SVBWMG14 : MERGE 出生, 死亡, 婚姻, 離婚, 共同使用 .....	54
7) SVBUSB14 : SORT 出生, 死亡, 婚姻, 離婚, 共同使用 .....	56
8) SVBWTB14 : 月別 地域別 總件數確認作業 出生, 死亡, 婚姻, 離婚, 共同使用 .....	57
9) SVBWCB14 : 最終 CORRECTION 出生, 死亡, 婚姻, 離婚, 共同使用...	59

10) SVBWS14 : 表 1 SUMMARY 出生, 死亡, 婚姻, 離婚, 合 .....	61
11) SVBWT14 : ADD UP .....	63
12) SVBWP14 : 總括表 1 .....	65
13) SVBWT214 : 表 2 SUMMARY 出生, 死亡, 婚姻, 離婚, 合 .....	68
14) SVBWD214 : ADD UP .....	73
15) SVBWP214 : 總括表 2 .....	75
16) SVBWSM14 : 表 3.4.5.6 SUMMARY .....	78
17) SVBWDL14 : ADD UP .....	83
18) SVBWTE14 : 表 3.4.5.6 .....	86
나. 死 亡 .....	95
1) SVDWTT14 : RIGHT JUSTIFY FILE .....	95
2) 出生 番號 2~番號 9 까지 共同使用 .....	96
3) SVDWSM14 : 表 7.8.9 SUMMARY .....	97
4) SVDWDL14 : ADD UP .....	101
5) SVDWTE14 : 表 7.8.9 .....	104
다. 婚 姻 .....	110
1) SVMWTT14 : RIGHT JUSTIFY FILE .....	110
2) 出生 番號 2~番號 9 까지 共同使用 .....	111
3) SVMWSM14 : 表 10.11.12.13 SUMMARY .....	112
4) SVMWDL14 : ADD UP .....	116
5) SVMWTE14 : 表 10.11.12.13 .....	120
라. 離 婚 .....	127
1) SVIWTT14 : RIGHT JUSTIFY FILE .....	127

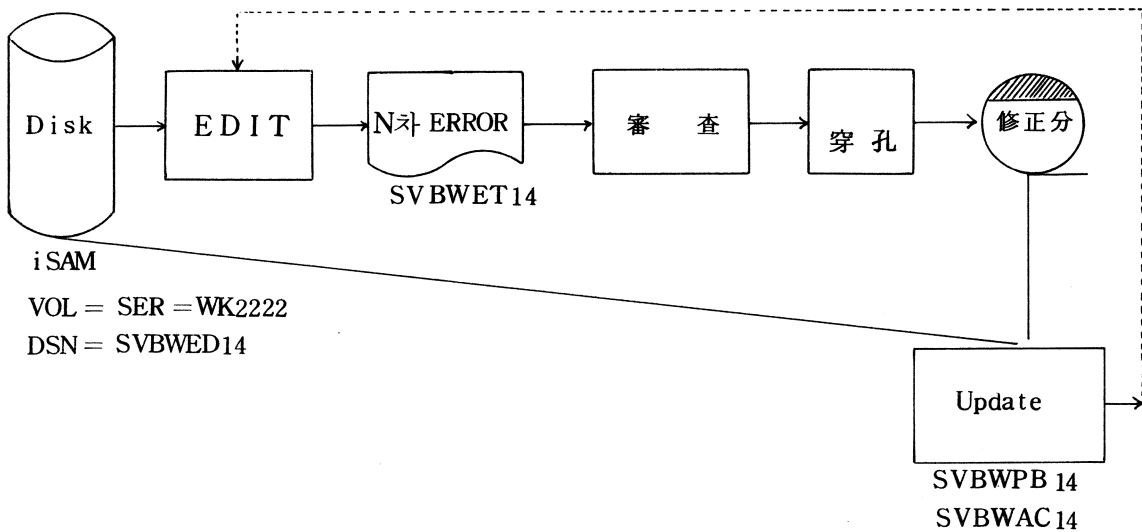
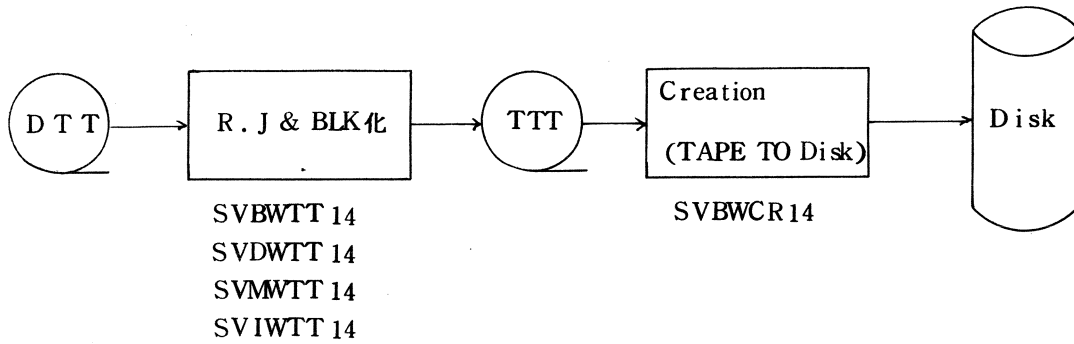
2) 出生 番號 2 ~ 番號 9 까지 共同使用 .....	128
3) SVIWSM14 : 表 14.15.16.17 SUMMARY .....	129
4) SVIWDL 14 : ADD UP .....	133
5) SVIWTE 14 : 表 14.15.16.17 .....	136
마. 死因分類 .....	142
1) SVDWST 14 : SELECTION .....	142
2) SVDWSR 14 : 總 括 表 .....	143
3) SVDWS 7 14 : 17分類를 위한 SUMMARY .....	150
4) SVDWT 7 14 : 表 17分類 .....	153
5) SVDWS 5 14 : 50分類를 위한 SUMMARY .....	161
6) SVDWT 5 14 : 表 50分類 .....	166
7) SVDWS 1 14 : 102分類를 위한 SUMMARY .....	175
8) SVDWT 0 14 : 表 102分類 .....	181
9) SVDWS 3 14 : 307分類를 위한 SUMMARY .....	186
10) SVDWT Z 14 : 表 307分類 .....	199
11) SVDWS 9 14 : 999分類를 위한 SUMMARY .....	209
12) SVDWT 9 14 : 表 999分類 .....	212
4. 人口動態申告 CRT 操作方法 및 用語解説 .....	215
5. 資料處理要求書作成方法 .....	225
6. 人口動態申告 結果表 .....	243

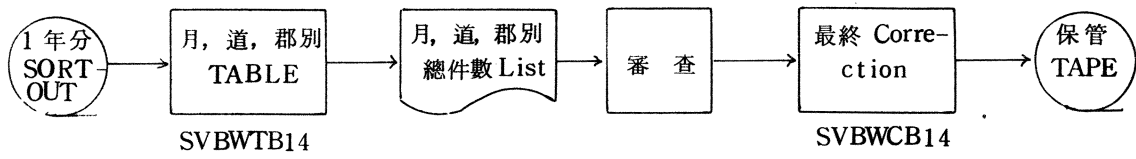
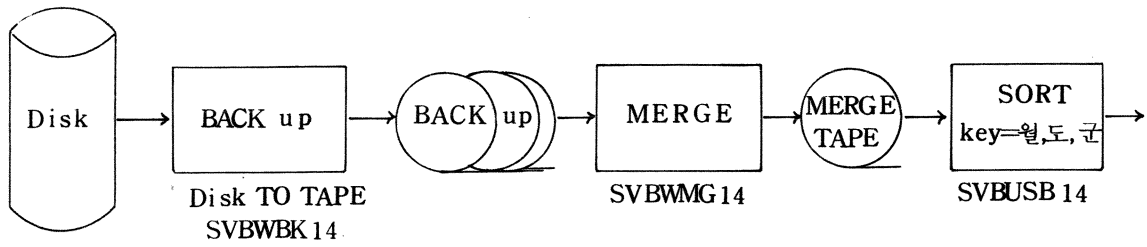
# 1. SYSTEM FLOW CHART

1) 當年分 作業

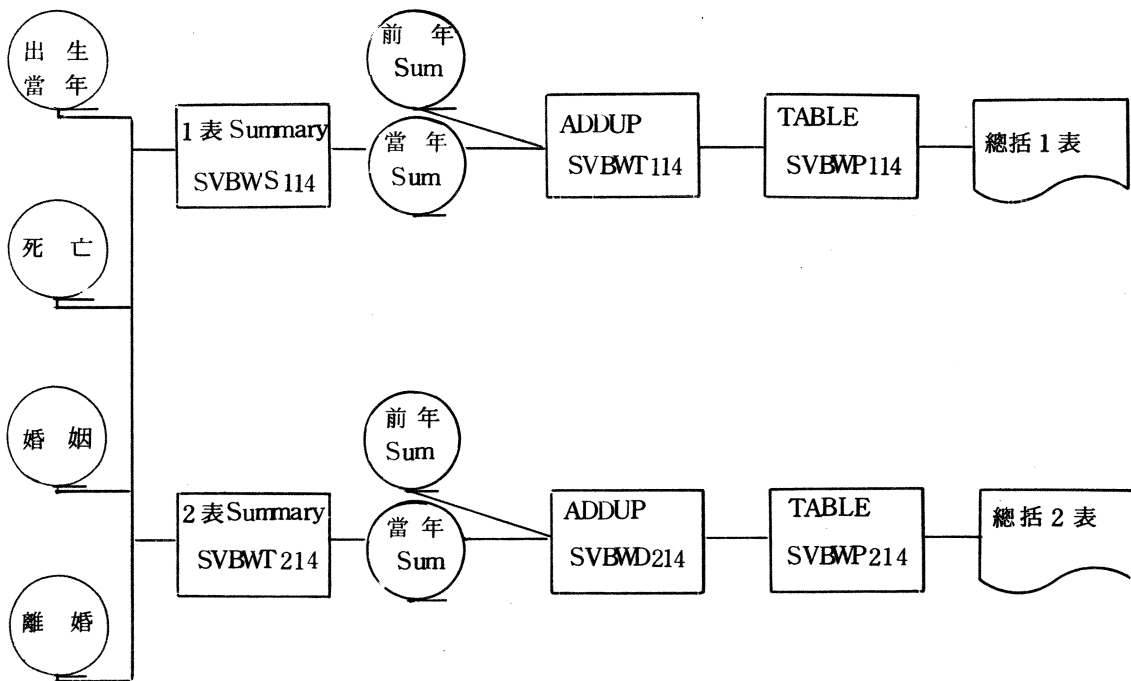


1. 出 生
2. 死 亡
3. 婚 姻
4. 離 婚

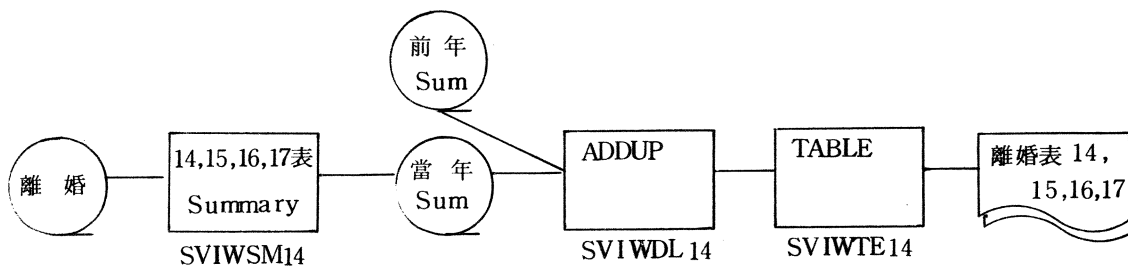
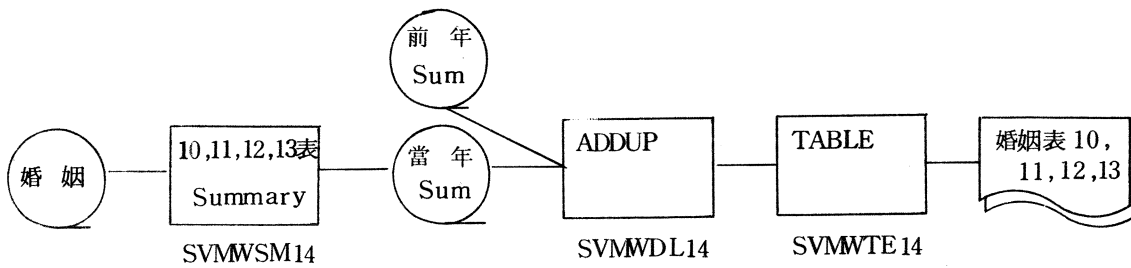
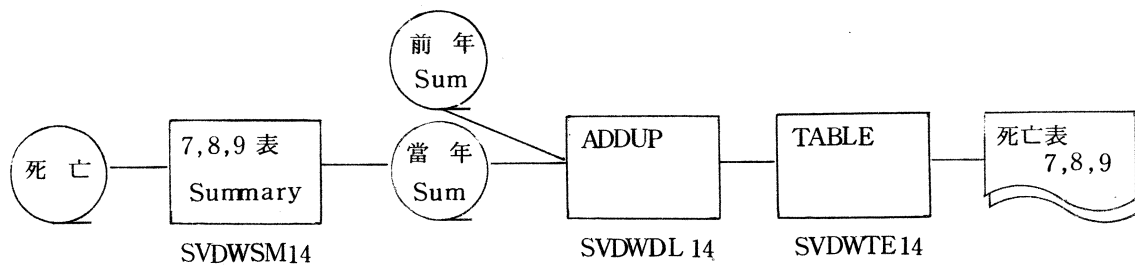
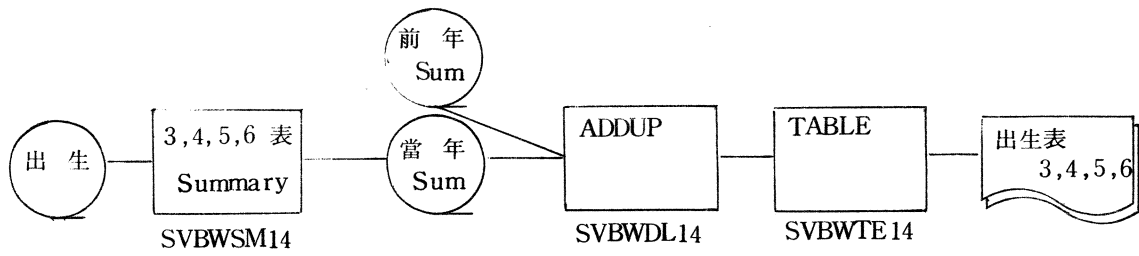




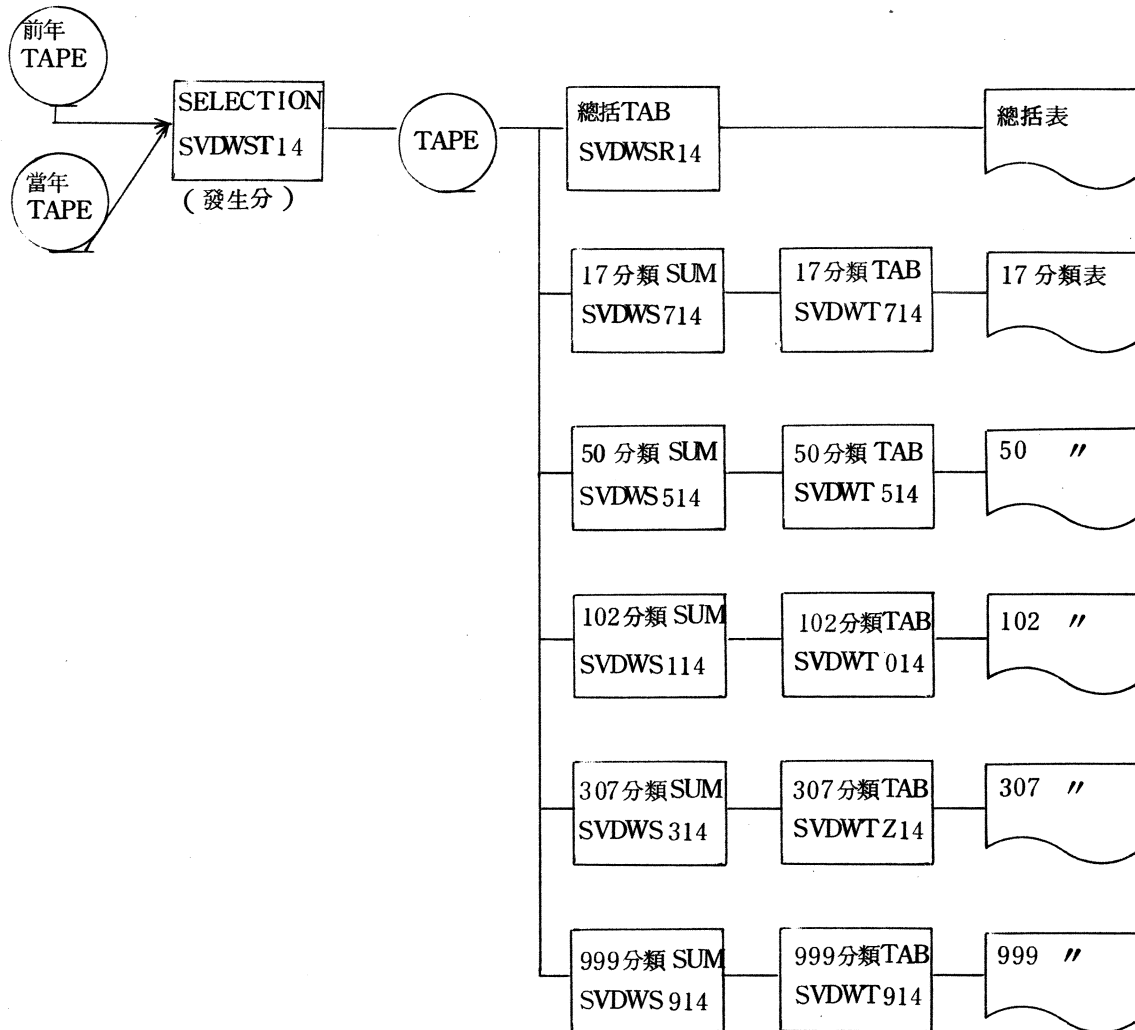
2) 10 個年 作業







3) 死因分類



## 2. 入力 DESIGN LAYOUT



# MULTIPLE TAPE LAYOUT FORM

인구동태신고    부호    구분 ~~10년분~~ <sup>VOL</sup> LBL    FILE LBL    · RL 40 BF 80 BS 3200 MFS

출생	출생부호 (1년)	신 고 지			출 생 자			부 모				모 의 총 출 산	t
		행정구역		일련번호	주소		출생연령	교육		동거기간			
		특별시·도	구시군동		도	군		년	월		t		
001	100	5	10	15	20	25	30	35	40				
사 망	사 망 부 호 (2)	사 망 자			사 망 년 월		직 인 동 태	사 망 년 월		1 개 월 이 하			
		주소		출생	년	월		t	1		유		
		도	군									년	월
101	200	5	10	15	20	25	30	35	40				
혼 인	혼 인 부 호 (4)	당 사 자			동 거 년 월		직 업	교 육		혼 인			
		주소		연 령	년	월		부	처		부	처	
		부	처										부
		도	군	도	군	부		처					
201	300	5	10	15	20	25	30	35	40				
이 혼	이 혼 부 호 (5)	당 사 자			이 혼 년 월		동 거 기 간	교 육		이 혼			
		주소		연 령	년	월		부	처		부	처	
		부	처										부
		도	군	도	군	부		처					
301	400	5	10	15	20	25	30	35	40				

### **3. SOURCE PROGRAM LIST**

```

//SVBWTT14 JOB TY=RUN=HOLD
//TTT EXEC COBOLC
//COB.SYSIN DD *
ID DIVISION.
PROGRAM-ID. TTT.
ENVIRONMENT DIVISION.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT T-F ASSIGN UT-S-TAPEIN.
    SELECT TA-F ASSIGN UT-S-TAPEOUT.
DATA DIVISION.
FILE SECTION.
FD T-F LABEL RECORD OMITTED.
01 T-R.
    03 AA PIC X(13).
    03 BB PIC X(45).
    03 CC PIC X(3).
    03 DD REDEFINES CC PIC 999.
    03 F PIC X(19).
FD TA-F BLOCK CONTAINS 40 RECORDS
LABEL RECORD STANDARD.
01 TA-R.
    03 TA PIC X(13).
    03 TC PIC 999.
    03 TB PIC X(45).
    03 F PIC X(19).
WORKING-STORAGE SECTION.
77 ECNT PIC 9(7) VALUE 0.
77 OCNT PIC 9(7) VALUE 0.
PROCEDURE DIVISION.
    OPEN INPUT T-F.
    OPEN OUTPUT TA-F.
    MOVE SPACE TO TA-R.
READ-RTN.
    READ T-F AT END GO TO END-RTN.
    ADD 1 TO ECNT.
    EXAMINE CC TALLYING ALL SPACE REPLACING BY 0.
    IF DD NOT NUMERIC GO TO MOVE-RTN.
    COMPUTE TC = DD / 10 ** TALLY.
MOVE-RTN.
    MOVE AA TO TA.
    MOVE BB TO TB.
    WRITE TA-R.
    MOVE SPACE TO TA-R.
    ADD 1 TO OCNT.
    GO TO READ-RTN.
END-RTN.
    DISPLAY " ECNT = " ECNT.
    DISPLAY " OCNT = " OCNT.
    CLOSE T-F TA-F.
    OPEN INPUT TA-F.
RD-RTN.
    READ TA-F AT END CLOSE TA-F STOP RUN.
    DISPLAY " " TA-R.
EX. PERFORM RD-RTN 2000 TIMES.
SVB00010
SVB00020
SVB00030
SVB00040
SVB00050
SVB00060
SVB00070
SVB00080
SVB00090
SVB00100
SVB00110
SVB00120
SVB00130
SVB00140
SVB00150
SVB00160
SVB00170
SVB00180
SVB00190
SVB00200
SVB00210
SVB00220
SVB00230
SVB00240
SVB00250
SVB00260
SVB00270
SVB00280
SVB00290
SVB00300
SVB00310
SVB00320
SVB00330
SVB00340
SVB00350
SVB00360
SVB00370
SVB00380
SVB00390
SVB00400
SVB00410
SVB00420
SVB00430
SVB00440
SVB00450
SVB00460
SVB00470
SVB00480
SVB00490
SVB00500
SVB00510
SVB00520
SVB00530
SVB00540
SVB00550

```

FILE: SVBWT14 RJTTT A1 VM/SP RELEASE 2.1 EXPRESS PUT8302+ SLU203

```

        CLOSE TA-F.
        STOP RUN.
/*
//GO.TAPEIN DD DSN=VBCHOI,UNIT=TAPE,DISP=OLD,VOL=SER=REEL01,
//          DCB=EROPT=SKF,LABEL=(,NL)
//          DD UNIT=AFF=TAPEIN,VOL=SER=REEL02,LABEL=(,NL)
//GO.TAPEOUT DD DSN=CHOI13,UNIT=TAPE,DISP=NEW,VOL=SER=SCRATCH
//SYSOUT DD SYSOUT=A
//
SVB00560
SVB00570
SVB00580
SVB00590
SVB00600
SVB00610
SVB00620
SVB00630
SVB00640
```



```

//SVBWCR14 JOB CLASS=V, TYPRUN=HOLD
//*      BDMI  WOL TAPE TO DISK EDITING  ***
//      EXEC PGM=IEHPRGM
//SYSPRINT DD SYSOUT=A
//DD3    DD UNIT=DISK, VOL=SER=WK2222, DISP=OLD
//SYSIN  DD *
  SCRATCH DSNAME=SVBWED14, VOL=DISK=WK2222
/*
//CREATION EXEC COBUCLG, PARM='NOSOURCE'
//COB.SYSIN DD *
  ID DIVISION.
  PROGRAM-ID.    BDMICR.
  ENVIRONMENT DIVISION.
  INPUT-OUTPUT SECTION.
  FILE-CONTROL.
    SELECT T-F ASSIGN UT-S-TAPEIN.
    SELECT DO-F ASSIGN DA-I-DISKOUT
      ACCESS MODE SEQUENTIAL
      RECORD KEY VBMDI.
    SELECT S-F ASSIGN DA-S-SORTWK.
  DATA DIVISION.
  FILE SECTION.
  FD T-F          BLOCK CONTAINS 40 RECORDS
                  LABEL RECORD STANDARD.
  01 T-R          PIC X(80).
  FD DO-F         BLOCK CONTAINS 40 RECORDS
                  LABEL RECORD STANDARD.
  01 DO-R.
    03 VBMDI      PIC X(16).
    03 F          PIC X(64).
  SD S-F          DATA RECORD S-R.
  01 S-R.
    03 VBMDI      PIC 9(16).
    03 VBD REDEFINES VBMDI.
      05 VB1 PIC X(13).
      05 VB2 PIC 999.
    03 BBM PIC X(38).
    03 F          PIC X(26).
  WORKING-STORAGE SECTION.
  77 ECNT PIC 9(7) VALUE 0.
  77 OCNT PIC 9(7) VALUE 0.
  77 BIGYO PIC 9(16) VALUE 0.
  PROCEDURE DIVISION.
    SORT S-F      ASCENDING KEY VBMDI
                  INPUT PROCEDURE INSORT
                  OUTPUT PROCEDURE OUTSORT.
    DISPLAY "    ECNT = " ECNT.
    DISPLAY "    OCNT = " OCNT.
    OPEN INPUT DO-F.
  RD-RTN.
    READ DO-F     AT END CLOSE DO-F STOP RUN.
    DISPLAY "    DO-R.
  PER-RTN.
    PERFORM RD-RTN 300 TIMES.
    CLOSE DO-F.

```

SVB00010  
SVB00020  
SVB00030  
SVB00040  
SVB00050  
SVB00060  
SVB00070  
SVB00080  
SVB00090  
SVB00100  
SVB00110  
SVB00120  
SVB00130  
SVB00140  
SVB00150  
SVB00160  
SVB00170  
SVB00180  
SVB00190  
SVB00200  
SVB00210  
SVB00220  
SVB00230  
SVB00240  
SVB00250  
SVB00260  
SVB00270  
SVB00280  
SVB00290  
SVB00300  
SVB00310  
SVB00320  
SVB00330  
SVB00340  
SVB00350  
SVB00360  
SVB00370  
SVB00380  
SVB00390  
SVB00400  
SVB00410  
SVB00420  
SVB00430  
SVB00440  
SVB00450  
SVB00460  
SVB00470  
SVB00480  
SVB00490  
SVB00500  
SVB00510  
SVB00520  
SVB00530  
SVB00540  
SVB00550

```

STOP RUN.
INSERT SECTION.
OPEN INPUT T-F.
READ-RTN.
    READ T-F AT END CLOSE T-F GO TO INSERT-EX.
    EXAMINE T-R REPLACING ALL SPACE BY 0.
MOVE-RTN.
    MOVE T-R TO S-R.
    RELEASE S-R.
    ADD 1 TO ECNT.
    GO TO READ-RTN.
INSERT-EX.
EXIT.
OUTSORT SECTION.
OPEN OUTPUT DO-F.
WRIT-RTN.
    RETURN S-F AT END GO TO OUTSORT-EX.
    IF VBDMI NOT NUMERIC GO TO WRIT-RTN.
    IF VBDMI = 0 GO TO WRIT-RTN.
    MOVE VBDMI TO BIGYO.
    GO TO MOV-RTN.
WRITE-RTN.
    RETURN S-F AT END CLOSE DO-F GO TO OUTSORT-EX.
    IF VBDMI NOT NUMERIC GO TO WRITE-RTN.
    IF VBDMI = 0 GO TO WRITE-RTN.
    IF VBDMI = BIGYO GO TO WRITE-RTN.
    MOVE VBDMI TO BIGYO.
MOV-RTN.
    MOVE S-R TO DO-R.
    WRITE DO-R INVALID KEY DISPLAY " KEY = " VBDMI.
    ADD 1 TO OCNT.
    GO TO WRITE-RTN.
OUTSORT-EX.
EXIT.
/*
//GO.TAPEIN DD DSN=SNOW14,UNIT=TAPE,DISP=OLD,VOL=SER=VBDMI1,
//          DCB=EROPT=SKP,LABEL=(2,BLF)
//GO.DISKOUT DD DSN=SVBWED14(INDEX),UNIT=DISK,DISP=(NEW,KEEP),
//          DCB=(DSORG=IS,KEYLEN=16,RKP=0,RECFM=FB),
//          VOL=SER=WK2222,SPACE=(CYL,(50,5))
//          DD DSN=SVBWED14(PRIME),UNIT=DISK,DISP=(NEW,KEEP),
//          DCB=DSORG=IS,
//          VOL=SER=WK2222,SPACE=(CYL,(175,5))
//          DD DSN=SVBWED14(OVFLOW),UNIT=DISK,DISP=(NEW,KEEP),
//          DCB=DSORG=IS,
//          VOL=SER=WK2222,SPACE=(CYL,30)
//GO.SORTWK DD UNIT=DISK,SPACE=(CYL,100)
//SORTWK01 DD UNIT=DISK,SPACE=(CYL,100)
//SORTWK02 DD UNIT=DISK,SPACE=(CYL,100)
//SORTWK03 DD UNIT=DISK,SPACE=(CYL,100)
//SYSOUT DD SYSOUT=A
//

```

SVB00560  
SVB00570  
SVB00580  
SVB00590  
SVB00600  
SVB00610  
SVB00620  
SVB00630  
SVB00640  
SVB00650  
SVB00660  
SVB00670  
SVB00680  
SVB00690  
SVB00700  
SVB00710  
SVB00720  
SVB00730  
SVB00740  
SVB00750  
SVB00760  
SVB00770  
SVB00780  
SVB00790  
SVB00800  
SVB00810  
SVB00820  
SVB00830  
SVB00840  
SVB00850  
SVB00860  
SVB00870  
SVB00880  
SVB00890  
SVB00900  
SVB00910  
SVB00920  
SVB00930  
SVB00940  
SVB00950  
SVB00960  
SVB00970  
SVB00980  
SVB00990  
SVB01000  
SVB01010  
SVB01020  
SVB01030  
SVB01040  
SVB01050  
SVB01060  
SVB01070

```

//SVBWET14 JOB CLASS=V, TYPRUN=HOLD SVB00010
//** BDMI DISK EDITING *** SVB00020
//***** BIRTH DATA EDITING ***** SVB00030
//DISKEDT EXEC COBUCLG, PARM='NOSOURCE' SVB00040
//COB.SYSIN DD * SVB00050
    ID DIVISION. SVB00060
    PROGRAM-ID. BDMIED. SVB00070
    ENVIRONMENT DIVISION. SVB00080
    INPUT-OUTPUT SECTION. SVB00090
    FILE-CONTROL. SVB00100
        SELECT D-F ASSIGN DA-I-DISKIN SVB00110
            ACCESS MODE SEQUENTIAL SVB00120
            RECORD KEY DEATH-A. SVB00130
        SELECT P-F ASSIGN UT-S-PROUT. SVB00140
    DATA DIVISION. SVB00150
    FILE SECTION. SVB00160
    FD D-F BLOCK CONTAINS 40 RECORDS SVB00170
        LABEL RECORD STANDARD. SVB00180
01 DB-R. SVB00190
    02 BIR. SVB00200
    03 BBU PIC X. SVB00210
    03 BYRM PIC 9(4). SVB00220
    03 BYRMM REDEFINES BYRM. SVB00230
        05 BSYP PIC 99. SVB00240
        05 BSMN PIC XX. SVB00250
            88 BSMNCD VALUES 01 02 03 04 05 06 07 08 09 10 11 12. SVB00260
        05 RBSMN REDEFINES BSMN PIC 99. SVB00270
    03 BHDO PIC XX. SVB00280
    03 BHGUN PIC XX. SVB00290
    03 BSBUN PIC 9(4). SVB00300
    03 BILBUN PIC 999. SVB00310
    02 BIR2. SVB00320
    03 BBDO PIC XX. SVB00330
    03 BBGUN PIC XX. SVB00340
    03 BJDO PIC XX. SVB00350
    03 BJGUN PIC XX. SVB00360
    03 BHJW PIC X. SVB00370
        88 BHJWCD VALUES 1 2. SVB00380
        88 JW VALUES 2. SVB00390
    03 BSEX PIC X. SVB00400
        88 BSEXCD VALUES 1 2. SVB00410
    03 BYRMN. SVB00420
        05 BYR PIC 99. SVB00430
        05 BMN PIC XX. SVB00440
            88 BMNCD VALUES 01 02 03 04 05 06 07 08 09 10 11 12. SVB00450
        05 RBMN REDEFINES BMN PIC 99. SVB00460
    03 BYEMN REDEFINES BYRMN PIC 9(4). SVB00470
    03 BPLS PIC X. SVB00480
        88 BPLSCD VALUES 1 2 3. SVB00490
    03 BFYRMN. SVB00500
        05 BFYR PIC 99. SVB00510
        05 RBFYR REDEFINES BFYR PIC XX. SVB00520
        05 BFMN PIC XX. SVB00530
            88 BFMNCD VALUES 01 02 03 04 05 06 07 08 09 10 11 12. SVB00540
        05 RBFMN REDEFINES BFMN PIC 99. SVB00550

```

03	BFYEMN	REDEFINES	BFYRMN	PIC 9(4).	SVB00560
03	BMYRMN.				SVB00570
	05	BMYR	PIC 99.		SVB00580
	05	RBMYR	REDEFINES	BMYR PIC XX.	SVB00590
	05	BMMN	PIC XX.		SVB00600
	88	BMMNCD	VALUES 01 02 03 04 05 06 07 08 09 10 11 12.		SVB00610
	05	RBMMN	REDEFINES	BMMN PIC 99.	SVB00620
03	BMYEMN	REDEFINES	BMYRMN	PIC 9(4).	SVB00630
03	BFOP	PIC XX.			SVB00640
	88	BFOPCD	VALUES 01 02 03 04 05 06 07		SVB00650
			10 11 12 13.		SVB00660
03	BMOP	PIC XX.			SVB00670
	88	BMOPCD	VALUES 01 02 03 04 05 06 07		SVB00680
			10 11 12 13.		SVB00690
03	BFET	PIC X.			SVB00700
	88	BFETCD	VALUES 1 2 3 4.		SVB00710
03	BMET	PIC X.			SVB00720
	88	BMETCD	VALUES 1 2 3 4.		SVB00730
03	BDYRMN.				SVB00740
	05	BDYR	PIC 99.		SVB00750
	05	RBDYR	REDEFINES	BDYR PIC XX.	SVB00760
	05	BDMN	PIC XX.		SVB00770
	88	BDMNCD	VALUES 01 02 03 04 05 06 07 08 09 10 11 12.		SVB00780
	05	RBDMN	REDEFINES	BDMN PIC 99.	SVB00790
03	BDYEMN	REDEFINES	BDYRMN	PIC 9(4).	SVB00800
03	BORN	PIC 99.			SVB00810
03	ALIVE	PIC 99.			SVB00820
03	DIE	PIC 99.			SVB00830
03	DBR	PIC 99.			SVB00840
03	PREG	PIC 99.			SVB00850
03	TAE	PIC X.			SVB00860
	88	TAECD	VALUES 1 2 3.		SVB00870
03	JOSAN	PIC X.			SVB00880
	88	JOSANCD	VALUES 1 2 3.		SVB00890
03	FAG	PIC 99.			SVB00900
03	MAG	PIC 99.			SVB00910
03	DONGFM	PIC 99.			SVB00920
02	F	PIC X(10).			SVB00930
02	BEL	PIC X(2).			SVB00940
02	F	PIC X(1).			SVB00950
01	DD-R.				SVB00960
02	DEATH-A.				SVB00970
	03	DBU.			SVB00980
		05	DBU	PIC 9.	SVB00990
		05	DYR	PIC 99.	SVB01000
	03	DMN	PIC XX.		SVB01010
	88	DMNCD	VALUES 01 02 03 04 05 06 07 08 09 10 11 12.		SVB01020
	03	DMMN	REDEFINES	DMN PIC 99.	SVB01030
	03	DHDO	PIC XX.		SVB01040
	03	DHGUN	PIC XX.		SVB01050
	03	DSBUN	PIC 9(4).		SVB01060
	03	DIL	PIC 9(3).		SVB01070
	03	DDIL	REDEFINES	DIL PIC XXX.	SVB01080
02	DEATH-B.				SVB01090
	03	DBDO	PIC XX.		SVB01100



	88 WMI VALUE		SVB01660
	020 021 023 024 050 060 073 085 100 102		SVB01670
	103 104 114 116 122 124.		SVB01680
	05 NOW PIC X.		SVB01690
	04 RWONIN REDEFINES WONIN PIC X(4).		SVB01700
	88 M3 VALUE 2334 THRU 2336 2364 THRU 2366		SVB01710
	4564 7525 THRU 7528.		SVB01720
	88 W3 VALUE 2331 THRU 2333 2360 THRU 2363		SVB01730
	7520 THRU 7524.		SVB01740
	88 WMJ VALUE 7980.		SVB01750
	04 W0NNN.		SVB01760
	05 FWON PIC X.		SVB01770
	05 W0NN PIC X(3).		SVB01780
	88 WMK VALUE 808 809 839 849 859 877 889 979.		SVB01790
	88 WML VALUE 950 THRU 959.		SVB01800
	05 F PIC X.		SVB01810
	03 AGYRMND.		SVB01820
	05 AGY PIC 9(3).		SVB01830
	05 AGW PIC 99.		SVB01840
	05 AGD PIC 99.		SVB01850
	02 F PIC X(16).		SVB01860
	02 DEL PIC XX.		SVB01870
	02 F PIC X.		SVB01880
01	DM-R.		SVB01890
	02 MAR-A.		SVB01900
	03 MBU PIC 9.		SVB01910
	03 MYRM.		SVB01920
	05 MMYR PIC 99.		SVB01930
	05 MMN PIC XX.		SVB01940
	88 MMNCD VALUES 01 02 03 04 05 06 07 08 09 10 11 12.		SVB01950
	05 RMMN REDEFINES MMN PIC 99.		SVB01960
	03 MYEM REDEFINES MYRM PIC 9(4).		SVB01970
	03 MHDO PIC XX.		SVB01980
	03 MHGUN PIC XX.		SVB01990
	03 MSBUN PIC 9(4).		SVB02000
	03 MIL PIC 9(3).		SVB02010
	02 MAR-B.		SVB02020
	03 MBHDO PIC XX.		SVB02030
	03 MBHG PIC XX.		SVB02040
	03 MBWDO PIC XX.		SVB02050
	03 MBWG PIC XX.		SVB02060
	03 MJHDO PIC XX.		SVB02070
	03 MJHG PIC XX.		SVB02080
	03 MJWDO PIC XX.		SVB02090
	03 MJWG PIC XX.		SVB02100
	03 MYRMON.		SVB02110
	05 MYR PIC 99.		SVB02120
	05 MMON PIC XX.		SVB02130
	88 MMONCD VALUES 01 02 03 04 05 06 07 08 09 10 11 12.		SVB02140
	05 RMMON REDEFINES MMON PIC 99.		SVB02150
	03 MYEMON REDEFINES MYRMON PIC 9(4).		SVB02160
	03 FYRMON.		SVB02170
	05 FYR PIC 99.		SVB02180
	05 FMON PIC XX.		SVB02190
	88 FMONCD VALUES 01 02 03 04 05 06 07 08 09 10 11 12.		SVB02200

	05 RFMON	REDEFINES FMON	PIC 99.	SVB02210
03	FYEMON	REDEFINES FYRMON	PIC 9(4).	SVB02220
03	DONGGW.			SVB02230
	05 DONGG	PIC 99.		SVB02240
	05 DONGW	PIC XX.		SVB02250
	88 DONGWCD	VALUES 01 02 03 04 05 06 07 08 09 10 11 12.		SVB02260
	05 RDONGW	REDEFINES DONGW	PIC 99.	SVB02270
03	DONGKW	REDEFINES DONGGW	PIC 9(4).	SVB02280
03	SHGJ	PIC X.		SVB02290
	88 SHGJCD	VALUES 1 2 3.		SVB02300
03	MPLS	PIC X.		SVB02310
	88 MPLSCD	VALUES 1 2 3.		SVB02320
03	MOPT	PIC XX.		SVB02330
	88 MOPTCD	VALUES 01 02 03 04 05 06 07 10 11 12 13.		SVB02340
				SVB02350
03	FOPT	PIC XX.		SVB02360
	88 FOPTCD	VALUES 01 02 03 04 05 06 07 10 11 12 13.		SVB02370
				SVB02380
03	MED	PIC X.		SVB02390
	88 MEDCD	VALUES 1 2 3 4.		SVB02400
03	FED	PIC X.		SVB02410
	88 FEDCD	VALUES 1 2 3 4.		SVB02420
03	HJM	PIC X.		SVB02430
	88 HJMCD	VALUES 1 2 3.		SVB02440
03	HJF	PIC X.		SVB02450
	88 HJFCD	VALUES 1 2 3.		SVB02460
03	MAGG	PIC 99.		SVB02470
03	FAGG	PIC 99.		SVB02480
03	F	PIC X(22).		SVB02490
01	DI-R.			SVB02500
02	DIVOR-A.			SVB02510
03	JBU	PIC 9.		SVB02520
	88 JCD	VALUE 5.		SVB02530
03	JSYR.			SVB02540
	05 JYR	PIC 99.		SVB02550
	05 JMN	PIC XX.		SVB02560
	88 JMNCD	VALUES 01 02 03 04 05 06 07 08 09 10 11 12.		SVB02570
	05 RJMN	REDEFINES JMN	PIC 99.	SVB02580
03	JSYE	REDEFINES JSYR	PIC 9(4).	SVB02590
03	JHDO	PIC XX.		SVB02600
03	JHGUN	PIC XX.		SVB02610
03	JSOK	PIC 9(4).		SVB02620
03	JIL	PIC 999.		SVB02630
02	DIVOR-B.			SVB02640
03	JBDO	PIC XX.		SVB02650
03	JBGUN	PIC XX.		SVB02660
03	JMDO	PIC XX.		SVB02670
03	JMGUN	PIC XX.		SVB02680
03	JFDO	PIC XX.		SVB02690
03	JFGUN	PIC XX.		SVB02700
03	MSYRWOL.			SVB02710
	05 MSYR	PIC 99.		SVB02720
	05 MSWOL	PIC XX.		SVB02730
	88 MSWOLCD	VALUES 01 02 03 04 05 06 07 08 09 10 11 12.		SVB02740
	05 RMSWOL	REDEFINES MSWOL	PIC 99.	SVB02750

03	MSYEWOL	REDEFINES MSYRWOL	PIC 9(4).	SVB02760	
03	FSYRWOL			SVB02770	
	05	FSYR	PIC 99.	SVB02780	
	05	FSWOL	PIC XX.	SVB02790	
	88	FSWOLCD	VALUES 01 02 03 04 05 06 07 08 09 10 11 12.	SVB02800	
	05	RFSWOL	REDEFINES FSWOL	PIC 99.	SVB02810
03	FSYEWOL	REDEFINES FSYRWOL	PIC 9(4).	SVB02820	
03	DONGNW			SVB02830	
	05	DONGN	PIC 99.	SVB02840	
	05	GUEW	PIC XX.	SVB02850	
	88	GUEWCD	VALUES 01 02 03 04 05 06 07 08 09 10 11 12.	SVB02860	
	05	RGUEW	REDEFINES GUEW	PIC 99.	SVB02870
03	DONGKNW	REDEFINES DONGNW	PIC 9(4).	SVB02880	
03	DIVYW			SVB02890	
	05	DIVY	PIC 99.	SVB02900	
	05	DIVW	PIC XX.	SVB02910	
	88	DIVWCD	VALUES 01 02 03 04 05 06 07 08 09 10 11 12.	SVB02920	
	05	RDIVW	REDEFINES DIVW	PIC 99.	SVB02930
03	DIVYWW	REDEFINES DIVYW	PIC 9(4).	SVB02940	
03	SUNGH	PIC X.		SVB02950	
	88	SUNGHCD	VALUES 1 2 3.	SVB02960	
03	DIVRES	PIC X.		SVB02970	
	88	DIVRESCD	VALUES 1 2 3 4 5 6.	SVB02980	
03	HED	PIC X.		SVB02990	
	88	HEDCD	VALUES 1 2 3 4.	SVB03000	
03	WED	PIC X.		SVB03010	
	88	WEDCD	VALUES 1 2 3 4.	SVB03020	
03	HAG	PIC 99.		SVB03030	
03	EAG	PIC 99.		SVB03040	
03	DONGK	PIC 99.		SVB03050	
03	F	PIC X(26).		SVB03060	
FD	P-F	LABEL RECORD OMITTED.		SVB03070	
01	P-R	PIC X(132).		SVB03080	
WORKING-STORAGE SECTION.				SVB03090	
77	I	PIC 999	VALUE 0.	SVB03100	
77	J	PIC 999	VALUE 0.	SVB03110	
77	K	PIC 999	VALUE 0.	SVB03120	
77	L	PIC 999	VALUE 0.	SVB03130	
77	M	PIC 999	VALUE 0.	SVB03140	
77	N	PIC 999	VALUE 0.	SVB03150	
77	ECNT	PIC 9(7)	VALUE 0.	SVB03160	
77	MCNT	PIC 9(7)	VALUE 0.	SVB03170	
77	DCNT	PIC 9(7)	VALUE 0.	SVB03180	
77	GCNT	PIC 9(7)	VALUE 0.	SVB03190	
77	OCNT	PIC 9(7)	VALUE 0.	SVB03200	
77	ERRCNT	PIC 9(7)	VALUE 0.	SVB03210	
77	SW	PIC 9	VALUE 0.	SVB03220	
77	SWD	PIC 9	VALUE 0.	SVB03230	
77	SWG	PIC 9	VALUE 0.	SVB03240	
77	B1	PIC 9(4)	VALUE 0.	SVB03250	
77	B2	PIC 999	VALUE 0.	SVB03260	
77	Y1	PIC 9999	VALUE 0.	SVB03270	
77	Y2	PIC 9999	VALUE 0.	SVB03280	
77	CNT	PIC 999	VALUE 1.	SVB03290	
77	DOB	PIC XX.		SVB03300	



77	GUNB	PIC	XX.	SVB03310
77	MBC	PIC	XX.	SVB03320
77	DOBK	PIC	XX.	SVB03330
77	GUNBK	PIC	XX.	SVB03340
01	COMPAREA.			SVB03350
	03	YMD	OCCURS 3.	SVB03360
		05	YY PIC 999.	SVB03370
		05	MM PIC 99.	SVB03380
		05	DD PIC 99.	SVB03390
**	NUMBER OF DD - GUN	**		SVB03400
01	GUNAT	PIC	X(40).	SVB03410
01	GUNTA	REDEFINES	GUNAT.	SVB03420
	03	GAT	OCCURS 20 PIC XX.	SVB03430
**	SINGO WOL	**		SVB03440
01	YEARV	PIC	XX.	SVB03450
**	BU HO	**		SVB03460
01	BUHO	PIC	9.	SVB03470
**	DO (DISTRICT DEFINE)	**		SVB03480
01	DOD	PIC	X(40).	SVB03490
01	DOF	REDEFINES	DOD.	SVB03500
	03	DON	OCCURS 20 PIC 99.	SVB03510
**	SEOUL GUN	**		SVB03520
01	SEL	PIC	X(80).	SVB03530
01	SEO	REDEFINES	SEL.	SVB03540
	03	SE	OCCURS 40 PIC XX.	SVB03550
**	BUSAN GUN	**		SVB03560
01	BUS	PIC	X(80).	SVB03570
01	BUN	REDEFINES	BUS.	SVB03580
	03	BS	OCCURS 40 PIC XX.	SVB03590
**	DAEGOO GUN	**		SVB03600
01	DAE	PIC	X(80).	SVB03610
01	GOO	REDEFINES	DAE.	SVB03620
	03	DG	OCCURS 40 PIC XX.	SVB03630
**	INCHEON GUN	**		SVB03640
01	INC	PIC	X(80).	SVB03650
01	HEON	REDEFINES	INC.	SVB03660
	03	IC	OCCURS 40 PIC XX.	SVB03670
**	KEOYNG KEE GUN	**		SVB03680
01	KEONG	PIC	X(80).	SVB03690
01	KEOG	REDEFINES	KEONG.	SVB03700
	03	KK	OCCURS 40 PIC XX.	SVB03710
**	KANG WON GUN	**		SVB03720
01	KANG	PIC	X(80).	SVB03730
01	KAG	REDEFINES	KANG.	SVB03740
	03	KW	OCCURS 40 PIC XX.	SVB03750
**	CHUNG BOOK GUN	**		SVB03760
01	CHUNB	PIC	X(80).	SVB03770
01	CHG	REDEFINES	CHUNB.	SVB03780
	03	CB	OCCURS 40 PIC XX.	SVB03790
**	CHUNG NAM GUN	**		SVB03800
01	CHUNAM	PIC	X(80).	SVB03810
01	CHNM	REDEFINES	CHUNAM.	SVB03820
	03	CN	OCCURS 40 PIC XX.	SVB03830
**	JEON BOOK	**		SVB03840
01	JEONB	PIC	X(80).	SVB03850

01	JEQB	REDEFINES JEONB.	SVB03860
	03	JB OCCURS 40 PIC XX.	SVB03870
**	JEON	NAM GUN **	SVB03880
01	JEONN	PIC X(80).	SVB03890
01	JEON	REDEFINES JEONN.	SVB03900
	03	JN OCCURS 40 PIC XX.	SVB03910
**	KEOYNG	BOOK **	SVB03920
01	KEOB	PIC X(80).	SVB03930
01	KEKB	REDEFINES KEOB.	SVB03940
	03	KB OCCURS 40 PIC XX.	SVB03950
**	KEONG	NAM **	SVB03960
01	KEONGN	PIC X(80).	SVB03970
01	KEON	REDEFINES KEONGN.	SVB03980
	03	KN OCCURS 40 PIC XX.	SVB03990
**	JE	JU GUN **	SVB04000
01	JEJUDO	PIC X(80).	SVB04010
01	JEJ	REDEFINES JEJUDO.	SVB04020
	03	JJ OCCURS 40 PIC XX.	SVB04030
01	SIGNALV	PIC 9.	SVB04040
01	CNT-PR.		SVB04050
	03	F PIC X VALUE SPACE.	SVB04060
	03	WOL-T PIC X(10).	SVB04070
	03	WOL-P PIC X(3).	SVB04080
	03	MCNT-P PIC Z(8)9.	SVB04090
	03	DO-T PIC X(10).	SVB04100
	03	DO-P PIC X(3).	SVB04110
	03	DCNT-P PIC Z(8)9.	SVB04120
	03	GUN-T PIC X(10).	SVB04130
	03	GUN-P PIC X(3).	SVB04140
	03	GCNT-P PIC Z(8)9.	SVB04150
01	COLDPY	PIC X(78) VALUE " 17.20...25...3	SVB04160
-		"0...35...40...45...50...55...60...65...70".	SVB04170
01	VB-PR	VALUE SPACES.	SVB04180
	02	VBBB.	SVB04190
	03	F PIC X.	SVB04200
	03	VB1 PIC X.	SVB04210
	03	VB2 PIC XX.	SVB04220
	03	VB3 PIC XXX.	SVB04230
	03	VB4 PIC XXX.	SVB04240
	03	VB5 PIC XXX.	SVB04250
	03	VB6 PIC XXXX.	SVB04260
	03	VB7 PIC XXXXX.	SVB04270
	02	VBBBB.	SVB04280
	03	VB8 PIC XX.	SVB04290
	03	VB9 PIC XX.	SVB04300
	03	VC1 PIC XX.	SVB04310
	03	VC2 PIC XX.	SVB04320
	03	VC3 PIC X.	SVB04330
	03	VC4 PIC X.	SVB04340
	03	VC5 PIC XX.	SVB04350
	03	VC6 PIC XX.	SVB04360
	03	VC7 PIC X.	SVB04370
	03	VC8 PIC XX.	SVB04380
	03	VC9 PIC XX.	SVB04390
	03	VD1 PIC XX.	SVB04400

03	VD2	PIC XX.	SVB04410
03	VD3	PIC XX.	SVB04420
03	VD4	PIC XX.	SVB04430
03	VD5	PIC X.	SVB04440
03	VD6	PIC X.	SVB04450
03	VD7	PIC XX.	SVB04460
03	VD8	PIC XX.	SVB04470
03	VD9	PIC XX.	SVB04480
03	VE1	PIC XX.	SVB04490
03	VE2	PIC XX.	SVB04500
03	VE3	PIC XX.	SVB04510
03	VE4	PIC XX.	SVB04520
03	VE5	PIC X.	SVB04530
03	VE6	PIC X.	SVB04540
03	VE7	PIC XX.	SVB04550
03	VE8	PIC XX.	SVB04560
03	VE9	PIC XX.	SVB04570
01	VB-D	VALUE SPACES.	SVB04580
03	F	PIC X.	SVB04590
03	DB1	PIC X.	SVB04600
03	DB2	PIC XX.	SVB04610
03	DB3	PIC XXX.	SVB04620
03	DB4	PIC XXX.	SVB04630
03	DB5	PIC XXX.	SVB04640
03	DB6	PIC XXXX.	SVB04650
03	DB7	PIC XXXXX.	SVB04660
03	DB8	PIC XX.	SVB04670
03	DB9	PIC XX.	SVB04680
03	DC1	PIC XX.	SVB04690
03	DC2	PIC XX.	SVB04700
03	DC3	PIC X.	SVB04710
03	DC4	PIC X.	SVB04720
03	DC5	PIC XX.	SVB04730
03	DC6	PIC XX.	SVB04740
03	DC7	PIC X.	SVB04750
03	DC8	PIC XX.	SVB04760
03	DC9	PIC XX.	SVB04770
03	DD1	PIC XX.	SVB04780
03	DD2	PIC XX.	SVB04790
03	DD3	PIC XX.	SVB04800
03	DD4	PIC XX.	SVB04810
03	DD5	PIC X.	SVB04820
03	DD6	PIC X.	SVB04830
03	DD7	PIC XX.	SVB04840
03	DD8	PIC XX.	SVB04850
03	DD9	PIC XX.	SVB04860
03	DE1	PIC XX.	SVB04870
03	DE2	PIC XX.	SVB04880
03	DE3	PIC XX.	SVB04890
03	DE4	PIC XX.	SVB04900
03	DE5	PIC X.	SVB04910
03	DE6	PIC X.	SVB04920
03	DE7	PIC XX.	SVB04930
03	DE8	PIC XX.	SVB04940
03	DE9	PIC XX.	SVB04950

01	VD-PR	VALUE SPACES.	SVB04960
	02	VDDD.	SVB04970
	03	F PIC X.	SVB04980
	03	DV1 PIC X.	SVB04990
	03	DV2 PIC XX.	SVB05000
	03	DV3 PIC XXX.	SVB05010
	03	DV4 PIC XXX.	SVB05020
	03	DV5 PIC XXX.	SVB05030
	03	DV6 PIC XXXX.	SVB05040
	03	DV7 PIC XXXXX.	SVB05050
	02	VDDDD.	SVB05060
	03	DV8 PIC XX.	SVB05070
	03	DV9 PIC XX.	SVB05080
	03	DW1 PIC XX.	SVB05090
	03	DW2 PIC XX.	SVB05100
	03	DW3 PIC X.	SVB05110
	03	DW4 PIC XXX.	SVB05120
	03	DW5 PIC XX.	SVB05130
	03	DW6 PIC XX.	SVB05140
	03	DW7 PIC XXX.	SVB05150
	03	DW8 PIC XX.	SVB05160
	03	DW9 PIC XX.	SVB05170
	03	DX1 PIC X.	SVB05180
	03	DX2 PIC XX.	SVB05190
	03	DX3 PIC X.	SVB05200
	03	DX4 PIC X.	SVB05210
	03	DX5 PIC X.	SVB05220
	03	DX6 PIC X(9).	SVB05230
	03	DX7 PIC X(3).	SVB05240
	03	DX8 PIC XX.	SVB05250
	03	DX9 PIC XX.	SVB05260
01	VD-D	VALUE SPACE.	SVB05270
	03	F PIC X.	SVB05280
	03	EV1 PIC X.	SVB05290
	03	EV2 PIC XX.	SVB05300
	03	EV3 PIC XXX.	SVB05310
	03	EV4 PIC XXX.	SVB05320
	03	EV5 PIC XXX.	SVB05330
	03	EV6 PIC XXXX.	SVB05340
	03	EV7 PIC XXXXX.	SVB05350
	03	EV8 PIC XX.	SVB05360
	03	EV9 PIC XX.	SVB05370
	03	EW1 PIC XX.	SVB05380
	03	EW2 PIC XX.	SVB05390
	03	EW3 PIC X.	SVB05400
	03	EW4 PIC XXX.	SVB05410
	03	EW5 PIC XX.	SVB05420
	03	EW6 PIC XX.	SVB05430
	03	EW7 PIC XXX.	SVB05440
	03	EW8 PIC XX.	SVB05450
	03	EW9 PIC XX.	SVB05460
	03	EX1 PIC X.	SVB05470
	03	EX2 PIC XX.	SVB05480
	03	EX3 PIC X.	SVB05490
	03	EX4 PIC X.	SVB05500

	03 EX5	PIC X.	SVB05510
	03 EX6	PIC X(9).	SVB05520
	03 EX7	PIC X(3).	SVB05530
01	VM-PR	VALUE SPACE.	SVB05540
	02 VMMM.		SVB05550
	03 F	PIC X.	SVB05560
	03 MA1	PIC X.	SVB05570
	03 MA2	PIC XX.	SVB05580
	03 MA3	PIC XXX.	SVB05590
	03 MA4	PIC XXX.	SVB05600
	03 MA5	PIC XXX.	SVB05610
	03 MA6	PIC XXXX.	SVB05620
	03 MA7	PIC XXXXX.	SVB05630
	02 VMMMM.		SVB05640
	03 MA8	PIC XX.	SVB05650
	03 MA9	PIC XX.	SVB05660
	03 MB1	PIC XX.	SVB05670
	03 MB2	PIC XX.	SVB05680
	03 MB3	PIC XX.	SVB05690
	03 MB4	PIC XX.	SVB05700
	03 MB5	PIC XX.	SVB05710
	03 MB6	PIC XX.	SVB05720
	03 MB7	PIC XX.	SVB05730
	03 MB8	PIC XX.	SVB05740
	03 MB9	PIC XX.	SVB05750
	03 MC1	PIC XX.	SVB05760
	03 MC2	PIC XX.	SVB05770
	03 MC3	PIC XX.	SVB05780
	03 MC4	PIC X.	SVB05790
	03 MC5	PIC X.	SVB05800
	03 MC6	PIC XX.	SVB05810
	03 MC7	PIC XX.	SVB05820
	03 MC8	PIC X.	SVB05830
	03 MC9	PIC X.	SVB05840
	03 MD1	PIC X.	SVB05850
	03 MD2	PIC X.	SVB05860
	03 MD3	PIC XX.	SVB05870
	03 MD4	PIC XX.	SVB05880
01	VM-D	VALUE SPACE.	SVB05890
	03 F	PIC X.	SVB05900
	03 FA1	PIC X.	SVB05910
	03 FA2FA3.		SVB05920
	05 FA2	PIC XX.	SVB05930
	05 FA3	PIC XXX.	SVB05940
	03 FA4	PIC XXX.	SVB05950
	03 FA5	PIC XXX.	SVB05960
	03 FA6	PIC XXXX.	SVB05970
	03 FA7	PIC XXXXX.	SVB05980
	03 FA8	PIC XX.	SVB05990
	03 FA9	PIC XX.	SVB06000
	03 FB1	PIC XX.	SVB06010
	03 FB2	PIC XX.	SVB06020
	03 FB3	PIC XX.	SVB06030
	03 FB4	PIC XX.	SVB06040
	03 FB5	PIC XX.	SVB06050

	03 FB6	PIC XX.	SVB06060
	03 FB7FB8.		SVB06070
	05 FB7	PIC XX.	SVB06080
	05 FB8	PIC XX.	SVB06090
	03 FB9FC1.		SVB06100
	05 FB9	PIC XX.	SVB06110
	05 FC1	PIC XX.	SVB06120
	03 FC2FC3.		SVB06130
	05 FC2	PIC XX.	SVB06140
	05 FC3	PIC XX.	SVB06150
	03 FC4	PIC X.	SVB06160
	03 FC5	PIC X.	SVB06170
	03 FC6	PIC XX.	SVB06180
	03 FC7	PIC XX.	SVB06190
	03 FC8	PIC X.	SVB06200
	03 FC9	PIC X.	SVB06210
	03 FD1	PIC X.	SVB06220
	03 FD2	PIC X.	SVB06230
	03 FD3	PIC XX.	SVB06240
	03 FD4	PIC XX.	SVB06250
01	VI-PR	VALUE SPACE.	SVB06260
	02 VIII.		SVB06270
	03 F	PIC X.	SVB06280
	03 CA1	PIC X.	SVB06290
	03 CA2	PIC XX.	SVB06300
	03 CA3	PIC XXX.	SVB06310
	03 CA4	PIC XXX.	SVB06320
	03 CA5	PIC XXX.	SVB06330
	03 CA6	PIC XXXX.	SVB06340
	03 CA7	PIC XXXXX.	SVB06350
	02 VIIII.		SVB06360
	03 CA8	PIC XX.	SVB06370
	03 CA9	PIC XX.	SVB06380
	03 CB1	PIC XX.	SVB06390
	03 CB2	PIC XX.	SVB06400
	03 CB3	PIC XX.	SVB06410
	03 CB4	PIC XX.	SVB06420
	03 CB5	PIC XX.	SVB06430
	03 CB6	PIC XX.	SVB06440
	03 CB7	PIC XX.	SVB06450
	03 CB8	PIC XX.	SVB06460
	03 CB9	PIC XX.	SVB06470
	03 CC1	PIC XX.	SVB06480
	03 CC2	PIC XX.	SVB06490
	03 CC3	PIC XX.	SVB06500
	03 CC4	PIC X.	SVB06510
	03 CC5	PIC X.	SVB06520
	03 CC6	PIC X.	SVB06530
	03 CC7	PIC X.	SVB06540
	03 CC8	PIC XX.	SVB06550
	03 CC9	PIC XX.	SVB06560
01	VI-D	VALUE SPACE.	SVB06570
	03 F	PIC X.	SVB06580
	03 GA1	PIC X.	SVB06590
	03 GA2	PIC XX.	SVB06600

03 GA3	PIC XXX.	SVB0660
03 GA4	PIC XXX.	SVB06620
03 GA5	PIC XXX.	SVB06630
03 GA6	PIC XXXX.	SVB06640
03 GA7	PIC XXXXX.	SVB06650
03 GA8	PIC XX.	SVB06660
03 GA9	PIC XX.	SVB06670
03 GB1	PIC XX.	SVB06680
03 GB2	PIC XX.	SVB06690
03 GB3	PIC XX.	SVB06700
03 GB4	PIC XX.	SVB06710
03 GB5	PIC XX.	SVB06720
03 GB6	PIC XX.	SVB06730
03 GB7	PIC XX.	SVB06740
03 GB8	PIC XX.	SVB06750
03 GB9	PIC XX.	SVB06760
03 GC1	PIC XX.	SVB06770
03 GC2	PIC XX.	SVB06780
03 GC3	PIC XX.	SVB06790
03 GC4	PIC X.	SVB06800
03 GC5	PIC X.	SVB06810
03 GC6	PIC X.	SVB06820
03 GC7	PIC X.	SVB06830
03 GC8	PIC XX.	SVB06840
03 GC9	PIC XX.	SVB06850
PROCEDURE DIVISION.		SVB06860
OPEN I-O D-F.		SVB06870
OPEN OUTPUT P-F.		SVB06880
MOVE SPACE TO P-R.		SVB06890
WRITE P-R AFTER POSITIONING 9.		SVB06900
MOVE SPACE TO P-R.		SVB06910
ACCEPT GUNAT.		SVB06920
ACCEPT YEARV.		SVB06930
ACCEPT BUHO.		SVB06940
ACCEPT DOD.		SVB06950
ACCEPT SEL.		SVB06960
ACCEPT BUS.		SVB06970
ACCEPT DAE.		SVB06980
ACCEPT INC.		SVB06990
ACCEPT KEONG.		SVB07000
ACCEPT KANG.		SVB07010
ACCEPT CHUNB.		SVB07020
ACCEPT CHUNAM.		SVB07030
ACCEPT JEONB.		SVB07040
ACCEPT JEONN.		SVB07050
ACCEPT KEOB.		SVB07060
ACCEPT KEONGN.		SVB07070
ACCEPT JEJUDO.		SVB07080
ACCEPT SIGNALV.		SVB07090
SEQ-RTN.		SVB07100
GO TO VB-ED VD-ED VD-ED VM-ED VI-ED	DEPENDING ON BUHO.	SVB07110
VB-ED.		SVB07120
READ D-F AT END GO TO END-RTN.		SVB07130
IF BEL = "***" GO TO VB-ED.		SVB07140
ADD 1 TO ECNT.		SVB07150

MOVE BSMN TO MBG.	SVB07160
MOVE BHDO TO DOBG.	SVB07170
MOVE BHGUN TO GUNBG.	SVB07180
GO TO VB-BI.	SVB07190
READ-VB.	SVB07200
READ D-F AT END GO TO VB-END.	SVB07210
IF BEL = "*" GO TO READ-VB.	SVB07220
ADD 1 TO ECNT.	SVB07230
VB-BI.	SVB07240
MOVE BUHO TO BBU.	SVB07250
MOVE YEARV TO BSYP.	SVB07260
MOVE BHDO TO DOB.	SVB07270
MOVE BHGUN TO GUNB.	SVB07280
IF NOT BSMNCD	SVB07290
MOVE "*" TO DB3	SVB07300
MOVE 0 TO SW.	SVB07310
PERFORM DO-RTN THRU DO-EX.	SVB07320
IF SWD = 1	SVB07330
MOVE "*" TO DB4	SVB07340
MOVE 0 TO SWD.	SVB07350
IF SWG = 1	SVB07360
MOVE "*" TO DB5	SVB07370
MOVE 0 TO SWG.	SVB07380
MOVE BBDO TO DOB.	SVB07390
MOVE BRGUN TO GUNB.	SVB07400
PERFORM DO-RTN THRU DO-EX.	SVB07410
IF SWD = 1	SVB07420
MOVE "*" TO DB8	SVB07430
MOVE 0 TO SWD.	SVB07440
IF SWG = 1	SVB07450
MOVE "*" TO DB9	SVB07460
MOVE 0 TO SWG.	SVB07470
MOVE BJDO TO DOB.	SVB07480
MOVE BJGUN TO GUNB.	SVB07490
PERFORM DO-RTN THRU DO-EX.	SVB07500
IF SWD = 1	SVB07510
MOVE "*" TO DC1	SVB07520
MOVE 0 TO SWD.	SVB07530
IF SWG = 1	SVB07540
MOVE "*" TO DC2	SVB07550
MOVE 0 TO SWG.	SVB07560
IF NOT BHJWCD	SVB07570
MOVE "*" TO DC3	SVB07580
MOVE 1 TO SW.	SVB07590
IF NOT BSEXCD	SVB07600
MOVE "*" TO DC4	SVB07610
MOVE 1 TO SW.	SVB07620
IF BYR NOT NUMERIC	SVB07630
MOVE "*" TO DC5	SVB07640
MOVE 1 TO SW.	SVB07650
IF NOT BMNCD	SVB07660
MOVE "*" TO DC6	SVB07670
MOVE 1 TO SW.	SVB07680
CPLS.	SVB07690
IF JW IF BPLS = "0" GO TO CFYR.	SVB07700



IF NOT BPLSCD		SVB07710
	MOVE "*" TO DC7	SVB07720
	MOVE 1 TO SW.	SVB07730
CFYR.		SVB07740
	IF JW IF RBFYR = "00" GO TO CFMN.	SVB07750
	IF BFYR NOT NUMERIC.	SVB07760
	MOVE "***" TO DC8	SVB07770
	MOVE 1 TO SW.	SVB07780
CFMN.		SVB07790
	IF JW IF BFMN = "00" GO TO CMYR.	SVB07800
	IF NOT BFMNCD	SVB07810
	MOVE "***" TO DC9	SVB07820
	MOVE 1 TO SW.	SVB07830
CMYR.		SVB07840
	IF JW IF RMYR = "00" GO TO CMMN.	SVB07850
	IF EMYR NOT NUMERIC	SVB07860
	MOVE "***" TO DD1	SVB07870
	MOVE 1 TO SW.	SVB07880
CMMN.		SVB07890
	IF JW IF BMMN = "00" GO TO CFOP.	SVB07900
	IF NOT BMMNCD	SVB07910
	MOVE "***" TO DD2	SVB07920
	MOVE 1 TO SW.	SVB07930
CFOP.		SVB07940
	IF JW IF BFOP = "00" GO TO CMOP.	SVB07950
	IF NOT BFOPCD	SVB07960
	MOVE "***" TO DD3	SVB07970
	MOVE 1 TO SW.	SVB07980
CMOP.		SVB07990
	IF JW IF BMOP = "00" GO TO CFET.	SVB08000
	IF NOT BMOPCD	SVB08010
	MOVE "***" TO DD4	SVB08020
	MOVE 1 TO SW.	SVB08030
CFET.		SVB08040
	IF JW IF BFET = "0" GO TO CMET.	SVB08050
	IF NOT BFETCD	SVB08060
	MOVE "*" TO DD5	SVB08070
	MOVE 1 TO SW.	SVB08080
CMET.		SVB08090
	IF JW IF BMET = "0" GO TO CDYR.	SVB08100
	IF NOT BMETCD	SVB08110
	MOVE "*" TO DD6	SVB08120
	MOVE 1 TO SW.	SVB08130
CDYR.		SVB08140
	IF JW IF RBDYR = "00" GO TO CDMN.	SVB08150
	IF RBYR NOT NUMERIC	SVB08160
	MOVE "***" TO DD7	SVB08170
	MOVE 1 TO SW.	SVB08180
CDMN.		SVB08190
	IF JW IF BDMN = "00" GO TO CBORN.	SVB08200
	IF NOT BDMNCD	SVB08210
	MOVE "***" TO DD8	SVB08220
	MOVE 1 TO SW.	SVB08230
CBORN.		SVB08240
	IF BORN NOT NUMERIC	SVB08250

```

        MOVE "***" TO DD9
        MOVE 1 TO SW GO TO SASAN-RTN.
IF ALIVE NOT NUMERIC
        MOVE "***" TO DE1
        MOVE 1 TO SW GO TO SASAN-RTN.
IF DIE NOT NUMERIC
        MOVE "***" TO DE2
        MOVE 1 TO SW.
        GO TO SASAN-RTN.
COMPUTE N = ALIVE + DIE.
IF BORN NOT = N
        MOVE "##" TO DD9
        MOVE 1 TO SW.
SASAN-RTN.
IF DBR NOT NUMERIC
        MOVE 0 TO DBR.
MOVE 10 TO PREG.
IF NOT TAECD
        MOVE 1 TO TAE.
IF SW NOT = 1 GO TO VB-COUNT.
VB-ERROR.
MOVE BBU TO VB1.
MOVE BSYR TO VB2.
MOVE BSMN TO VB3.
MOVE BHDD TO VB4.
MOVE BHGUN TO VB5.
MOVE BSBUN TO VB6.
MOVE BILBUN TO VB7.
MOVE BIR2 TO VB8.
WRITE P-R FROM COLDPY AFTER POSITIONING 2.
MOVE SPACE TO P-R.
WRITE P-R FROM VB-PR AFTER POSITIONING 2.
MOVE SPACE TO P-R.
WRITE P-R FROM VB-D AFTER POSITIONING 1.
MOVE SPACE TO P-R VB-PR VB-D.
MOVE 0 TO SW.
ADD 1 TO ERRCNT.
GO TO READ-VB.
VB-COUNT.
* IF BFYEMN > BYEMN MOVE "OV" TO DC8
* MOVE "ER" TO DC9
* GO TO VB-ERROR.
* IF BMYEMN > BYEMN MOVE "OV" TO DD1
* MOVE "ER" TO DD2
* GO TO VB-ERROR.
* IF BDYEMN > BYEMN MOVE "OV" TO DD7
* MOVE "ER" TO DD8
* GO TO VB-ERROR.
COM-F.
IF RBFYR = "00" OR BFMN = "00" GO TO COM-M.
IF RBFMN > RBMN COMPUTE FAG = BYR - BFYR - 1
ELSE COMPUTE FAG = BYR - BFYR.
COM-M.
IF RBMYR = "00" OR BMMN = "00" GO TO COM-D.
IF RBMMN > RBMN COMPUTE MAG = BYR - BMYR - 1

```

SVB08260  
SVB08270  
SVB08280  
SVB08290  
SVB08300  
SVB08310  
SVB08320  
SVB08330  
SVB08340  
SVB08350  
SVB08360  
SVB08370  
SVB08380  
SVB08390  
SVB08400  
SVB08410  
SVB08420  
SVB08430  
SVB08440  
SVB08450  
SVB08460  
SVB08470  
SVB08480  
SVB08490  
SVB08500  
SVB08510  
SVB08520  
SVB08530  
SVB08540  
SVB08550  
SVB08560  
SVB08570  
SVB08580  
SVB08590  
SVB08600  
SVB08610  
SVB08620  
SVB08630  
SVB08640  
SVB08650  
SVB08660  
SVB08670  
SVB08680  
SVB08690  
SVB08700  
SVB08710  
SVB08720  
SVB08730  
SVB08740  
SVB08750  
SVB08760  
SVB08770  
SVB08780  
SVB08790  
SVB08800

```

ELSE COMPUTE MAG = BYR - BMYR. SVB08810
COM-D. SVB08820
IF RBDYR = "00" OR BDMN = "00" GO TO COM-FMD. SVB08830
IF RBDMN > RBMN COMPUTE DONGFM = BYR - BDYR - 1 SVB08840
ELSE COMPUTE DONGFM = BYR - BDYR. SVB08850
COM-FMD. SVB08860
IF BFMN = "00" OR BMMN = "00" OR BDMN = "00" OR
RBFYR = "00" OR RBMYR = "00" OR RBDYR = "00" GO TO REWRT. SVB08870
IF FAG < 15 MOVE "%1" TO DE7 MOVE 1 TO SW. SVB08880
IF MAG < 15 OR MAG > 49 MOVE "%2" TO DE8 SVB08890
GO TO VB-ERROR. SVB08910
IF MAG < 20 IF BORN > 4 MOVE "%3" TO DE8 DD9 SVB08920
GO TO VB-ERROR. SVB08930
IF MAG < 25 IF BORN > 6 MOVE "%4" TO DE8 DD9 SVB08940
GO TO VB-ERROR. SVB08950
IF MAG < 30 IF BORN > 8 MOVE "%5" TO DE8 DD9 SVB08960
GO TO VB-ERROR. SVB08970
IF MAG < 35 IF BORN > 9 MOVE "%6" TO DE8 DD9 SVB08980
GO TO VB-ERROR. SVB08990
IF DONGFM > FAG MOVE "%7" TO DE9 SVB09000
MOVE 1 TO SW. SVB09010
IF SW = 1 GO TO VB-ERROR. SVB09020
REWRT. SVB09030
REWRITE DB-R. SVB09040
ADD 1 TO OCNT. SVB09050
GO TO READ-VB. SVB09060
VB-END. SVB09070
GO TO END-RTN. SVB09080
VD-ED. SVB09090
READ D-F AT END GO TO END-RTN. SVB09100
IF DEL = "***" GO TO VD-ED. SVB09110
ADD 1 TO ECNT. SVB09120
MOVE DMN TO MBG. SVB09130
MOVE DHDO TO DOBG. SVB09140
MOVE DHGUN TO GUNBG. SVB09150
GO TO VD-BI. SVB09160
READ-VD. SVB09170
READ D-F AT END GO TO VD-END. SVB09180
IF DEL = "***" GO TO READ-VD. SVB09190
ADD 1 TO ECNT. SVB09200
VD-BI. SVB09210
MOVE BUHO TO DBU. SVB09220
MOVE YEARV TO DYR. SVB09230
MOVE DHDO TO DOB. SVB09240
MOVE DHGUN TO GUNB. SVB09250
IF NOT DMNCD SVB09260
MOVE "***" TO EV3 SVB09270
MOVE 0 TO SW. SVB09280
PERFORM DO-RTN THRU DO-EX. SVB09290
IF SWD = 1 SVB09300
MOVE "***" TO EV4 SVB09310
MOVE 0 TO SWD. SVB09320
IF SWG = 1 SVB09330
MOVE "***" TO EV5 SVB09340
MOVE 0 TO SWG. SVB09350

```

MOVE DBDO TO DOB.		SVB09360
MOVE DBGUN TO GUNB.		SVB09370
PERFORM DO-RTN THRU DO-EX.		SVB09380
IF SWD = 1		SVB09390
	MOVE "***" TO EV8	SVB09400
	MOVE 0 TO SWD.	SVB09410
IF SWG = 1		SVB09420
	MOVE "***" TO EV9	SVB09430
	MOVE 0 TO SWG.	SVB09440
	MOVE 0 TO SWG.	SVB09450
MOVE DJDO TO DOB.		SVB09460
MOVE DJGUN TO GUNB.		SVB09470
PERFORM DO-RTN THRU DO-EX.		SVB09480
IF SWD = 1		SVB09490
	MOVE "***" TO EW1	SVB09500
	MOVE 0 TO SWD.	SVB09510
IF SWG = 1		SVB09520
	MOVE "***" TO EW2	SVB09530
	MOVE 0 TO SWG.	SVB09540
	MOVE 0 TO SWG.	SVB09550
IF NOT DSEXCD		SVB09560
	MOVE "*" TO EW3	SVB09570
	MOVE 1 TO SW.	SVB09580
IF DBYR NOT NUMERIC		SVB09590
	MOVE "****" TO EW4	SVB09600
	MOVE 1 TO SW.	SVB09610
IF NOT DBMNC		SVB09620
	MOVE "***" TO EW5	SVB09630
	MOVE 1 TO SW.	SVB09640
IF DBDYY NOT NUMERIC		SVB09650
	MOVE "***" TO EW6	SVB09660
	MOVE 1 TO SW	SVB09670
	GO TO DDYR-RTN.	SVB09680
IF DBDYY = 0		SVB09690
	MOVE "***" TO EW6	SVB09700
	MOVE 1 TO SW.	SVB09710
DDYR-RTN.		SVB09720
IF DDYR NOT NUMERIC		SVB09730
	MOVE "****" TO EW7	SVB09740
	MOVE 1 TO SW.	SVB09750
IF NOT DDMNC		SVB09760
	MOVE "***" TO EW8	SVB09770
	MOVE 1 TO SW.	SVB09780
IF DDDYY NOT NUMERIC		SVB09790
	MOVE "***" TO EW9	SVB09800
	MOVE 1 TO SW	SVB09810
	GO TO DPLS-RTN.	SVB09820
IF DDDYY = 0		SVB09830
	MOVE "***" TO EW9	SVB09840
	MOVE 1 TO SW.	SVB09850
DPLS-RTN.		SVB09860
IF NOT DPLSCD		SVB09870
	MOVE "*" TO EX1	SVB09880
	MOVE 1 TO SW.	SVB09890
IF NOT DOPTCD		SVB09900

	MOVE "***" TO EX2	SVB09910
	MOVE 1 TO SW.	SVB09920
IF NOT DJDCD		SVB09930
	MOVE "*" TO EX3	SVB09940
	MOVE 1 TO SW.	SVB09950
IF NOT DMSTCD		SVB09960
	MOVE "*" TO EX4	SVB09970
	MOVE 1 TO SW.	SVB09980
IF NOT DEDTCD		SVB09990
	MOVE "*" TO EX5	SVB10000
	MOVE 1 TO SW.	SVB10010
EXAMINE DEACA REPLACING ALL SPACE BY 0.		SVB10020
IF WON NOT NUMERIC		SVB10030
	MOVE "***" TO EX6	SVB10040
	MOVE 1 TO SW	SVB10050
	GO TO VD-ERR.	SVB10060
IF (W1 OR W2 OR W3) AND DSEX = 1		SVB10070
	MOVE "CHANGE SEX" TO EX6	SVB10080
	MOVE 1 TO SW	SVB10090
	GO TO VD-ERR.	SVB10100
IF (M1 OR M2 OR M3) AND DSEX = 2		SVB10110
	MOVE "CHANGE SEX" TO EX6	SVB10120
	MOVE 1 TO SW.	SVB10130
VD-ERR.		SVB10140
IF SW = 0	GO TO VD-COUNT.	SVB10150
VD-ERROR.		SVB10160
MOVE DDBU	TO DV1.	SVB10170
MOVE DYR	TO DV2.	SVB10180
MOVE DMN	TO DV3.	SVB10190
MOVE DHDO	TO DV4.	SVB10200
MOVE DHGUN	TO DV5.	SVB10210
MOVE DSBUN	TO DV6.	SVB10220
MOVE DIL	TO DV7.	SVB10230
MOVE DEATH-B	TO VDDDD.	SVB10240
WRITE P-R FROM COLDPY AFTER POSITIONING 2.		SVB10250
MOVE SPACE TO P-R.		SVB10260
WRITE P-R FROM VD-PR AFTER POSITIONING 2.		SVB10270
MOVE SPACE TO P-R.		SVB10280
WRITE P-R FROM VD-D AFTER POSITIONING 1.		SVB10290
MOVE SPACE TO P-R VD-PR	VD-D.	SVB10300
MOVE 0 TO SW.		SVB10310
ADD 1 TO ERRCNT.		SVB10320
GO TO READ-VD.		SVB10330
VD-COUNT.		SVB10340
MOVE DDYRMND	TO YMD(1).	SVB10350
MOVE DBYRMND	TO YMD(2).	SVB10360
IF DD(1) < DD(2)	COMPUTE DD(3) = DD(1) + 30 - DD(2)	SVB10370
	COMPUTE MM(1) = MM(1) - 1	SVB10380
ELSE	COMPUTE DD(3) = DD(1) - DD(2).	SVB10390
IF MM(1) < MM(2)	COMPUTE MM(3) = MM(1) + 12 - MM(2)	SVB10400
	COMPUTE YY(1) = YY(1) - 1	SVB10410
ELSE	COMPUTE MM(3) = MM(1) - MM(2).	SVB10420
COMPUTE YY(3) = YY(1) - YY(2).		SVB10430
IF YY(3) NOT = 0	MOVE 0 TO MM(3) DD(3)	SVB10440
ELSE IF MM(3) NOT = 0	MOVE 0 TO YY(3) DD(3)	SVB10450

```

ELSE MOVE 0 TO YY(3) MM(3). SVB10460
MOVE YMD(3) TO AGYRMD. SVB10470
MOVE ZEROS TO COMPAREA. SVB10480
*
IF RWONIN = "3005" IF FWON = "1" IF WONN = "958" SVB10490
MOVE "9599" TO RWONIN. SVB10500
IF DDYR < 980 GO TO VD-OK. SVB10510
COMPUTE Y1 = DYR + 1900. SVB10520
COMPUTE Y2 = DDYR + 1000. SVB10530
IF Y1 < Y2 MOVE "###" TO EV2 EW7 MOVE 1 TO SW. SVB10540
IF WON > 799 IF WONN = "000" SVB10550
MOVE "###" TO EX6 MOVE 1 TO SW. SVB10560
IF WON < 800 IF WONN NOT = "000" SVB10570
MOVE "###" TO EX6 MOVE 1 TO SW. SVB10580
IF WMB IF AGY > 14 IF AGY < 50 IF DSEX = 2 NEXT SENTENCE SVB10590
ELSE MOVE "BBB" TO EX6 EX7 MOVE 1 TO SW. SVB10600
IF WMC IF AGY > 0 MOVE "CCC" TO EX6 EX7 MOVE 1 TO SW. SVB10610
IF WMD IF AGY < 50 MOVE "DDD" TO EX6 EX7 MOVE 1 TO SW. SVB10620
IF WME IF AGY < 15 MOVE "EEE" TO EX6 EX7 MOVE 1 TO SW. SVB10630
IF WMF IF AGY > 49 MOVE "FFF" TO EX6 EX7 MOVE 1 TO SW. SVB10640
IF WMG IF AGY < 50 MOVE "GGG" TO EX6 EX7 MOVE 1 TO SW. SVB10650
IF WMH MOVE "HHH" TO EX6 EX7 MOVE 1 TO SW. SVB10660
IF WMI MOVE "III" TO EX6 EX7 MOVE 1 TO SW. SVB10670
IF WMJ IF AGY > 9 MOVE "JJJ" TO EX6 EX7 MOVE 1 TO SW. SVB10680
IF WMK MOVE "KKK" TO EX6 EX7 MOVE 1 TO SW. SVB10690
IF WML IF AGY < 9 MOVE "LLL" TO EX6 EX7 MOVE 1 TO SW. SVB10700
IF AGY < 14 IF DOPT = "11" OR "10" NEXT SENTENCE SVB10710
ELSE MOVE "###" TO EX2 EX7 MOVE 1 TO SW. SVB10720
IF SW = 1 GO TO VD-ERROR. SVB10730
VD-OK. SVB10740
REWRITE DD-R. SVB10750
ADD 1 TO OCNT. SVB10760
GO TO READ-VD. SVB10770
VD-YR. SVB10780
IF DDMNN < DBMNN COMPUTE AGY = DDYR - DBYR - 1 SVB10790
ELSE COMPUTE AGY = DDYR - DBYR. SVB10800
MOVE 0 TO AGW AGD. SVB10810
VD-MN. SVB10820
IF DDDYY < DBDYY COMPUTE AGW = DDMNN - DBMNN - 1 SVB10830
ELSE COMPUTE AGW = DDMNN - DBMNN. SVB10840
MOVE 0 TO AGY AGD. SVB10850
VD-END. SVB10860
GO TO END-RTN. SVB10870
VM-ED. SVB10880
READ D-F AT END GO TO END-RTN. SVB10890
ADD 1 TO ECNT. SVB10900
MOVE MMN TO MBG. SVB10910
MOVE MHDO TO DOBG. SVB10920
MOVE MHGUN TO GUNBG. SVB10930
GO TO VM-BI. SVB10940
READ-VM. SVB10950
READ D-F AT END GO TO VM-END. SVB10960
ADD 1 TO ECNT. SVB10970
VM-BI. SVB10980
MOVE BUHO TO MBU. SVB10990
SVB11000

```

MOVE YEARV TO MMYR.	SVB11010
MOVE MHDO TO DOB.	SVB11020
MOVE MHGUN TO GUNB.	SVB11030
IF NOT MMNCD	SVB11040
MOVE "***" TO FA3	SVB11050
MOVE 0 TO SW.	SVB11060
PERFORM DO-RTN THRU DO-EX.	SVB11070
IF SWD = 1	SVB11080
MOVE "***" TO FA4	SVB11090
MOVE 0 TO SWD.	SVB11100
IF SWG = 1	SVB11110
MOVE "***" TO FA5	SVB11120
MOVE 0 TO SWG.	SVB11130
MOVE MBHDO TO DOB.	SVB11140
MOVE MBHG TO GUNB.	SVB11150
PERFORM DO-RTN THRU DO-EX.	SVB11160
IF SWD = 1	SVB11170
MOVE "***" TO FAB	SVB11180
MOVE 0 TO SWD.	SVB11190
IF SWG = 1	SVB11200
MOVE "***" TO FA9	SVB11210
MOVE 0 TO SWG.	SVB11220
MOVE MBWDO TO DOB.	SVB11230
MOVE MBWG TO GUNB.	SVB11240
PERFORM DO-RTN THRU DO-EX.	SVB11250
IF SWD = 1	SVB11260
MOVE "***" TO FB1	SVB11270
MOVE 0 TO SWD.	SVB11280
IF SWG = 1	SVB11290
MOVE "***" TO FB2	SVB11300
MOVE 0 TO SWG.	SVB11310
MOVE MJHDO TO DOB.	SVB11320
MOVE MJHG TO GUNB.	SVB11330
PERFORM DO-RTN THRU DO-EX.	SVB11340
IF SWD = 1	SVB11350
MOVE "***" TO FB3	SVB11360
MOVE 0 TO SWD.	SVB11370
IF SWG = 1	SVB11380
MOVE "***" TO FB4	SVB11390
MOVE 0 TO SWG.	SVB11400
MOVE MJWDO TO DOB.	SVB11410
MOVE MJWG TO GUNB. PERFORM DO-RTN THRU DO-EX.	SVB11420
IF SWD = 1	SVB11430
MOVE "***" TO FB5	SVB11440
MOVE 0 TO SWD.	SVB11450
IF SWG = 1	SVB11460
MOVE "***" TO FB6	SVB11470
MOVE 0 TO SWG.	SVB11480
IF MYR NOT NUMERIC	SVB11490
MOVE "***" TO FB7	SVB11500
MOVE 1 TO SW.	SVB11510
IF NOT MMONCD	SVB11520
MOVE "***" TO FB8	SVB11530
MOVE 1 TO SW.	SVB11540
IF FYR NOT NUMERIC	SVB11550

	MOVE "***" TO FB9	SVB11560
	MOVE 1 TO SW.	SVB11570
IF NOT FMONCD		SVB11580
	MOVE "***" TO FC1	SVB11590
	MOVE 1 TO SW.	SVB11600
IF DONGG NOT NUMERIC		SVB11610
	MOVE "***" TO FC2	SVB11620
	MOVE 1 TO SW.	SVB11630
IF NOT DONGWCD		SVB11640
	MOVE "***" TO FC3	SVB11650
	MOVE 1 TO SW.	SVB11660
IF NOT SHGJCD		SVB11670
	MOVE "*" TO FC4	SVB11680
	MOVE 1 TO SW.	SVB11690
IF NOT MPLSCD		SVB11700
	MOVE "*" TO FC5	SVB11710
	MOVE 1 TO SW.	SVB11720
IF NOT MOPTCD		SVB11730
	MOVE "***" TO FC6	SVB11740
	MOVE 1 TO SW.	SVB11750
IF NOT FOPTCD		SVB11760
	MOVE "***" TO FC7	SVB11770
	MOVE 1 TO SW.	SVB11780
IF NOT MEDCD		SVB11790
	MOVE "*" TO FC8	SVB11800
	MOVE 1 TO SW.	SVB11810
IF NOT FEDCD		SVB11820
	MOVE "*" TO FC9	SVB11830
	MOVE 1 TO SW.	SVB11840
IF NOT HJMCD		SVB11850
	MOVE "*" TO FD1	SVB11860
	MOVE 1 TO SW.	SVB11870
IF NOT HJFCD		SVB11880
	MOVE "*" TO FD2	SVB11890
	MOVE 1 TO SW.	SVB11900
IF SW = 0	GO TO VM-COUNT.	SVB11910
VM-ERROR.		SVB11920
	MOVE MBU TO MA1.	SVB11930
	MOVE MMYR TO MA2.	SVB11940
	MOVE MMN TO MA3.	SVB11950
	MOVE MHDO TO MA4.	SVB11960
	MOVE MHGUN TO MA5.	SVB11970
	MOVE MSBUN TO MA6.	SVB11980
	MOVE MIL TO MA7.	SVB11990
	MOVE MAR-B TO VMMMM.	SVB12000
	WRITE P-R FROM COLDPY AFTER POSITIONING 2.	SVB12010
	MOVE SPACE TO P-R.	SVB12020
	WRITE P-R FROM VM-PR AFTER POSITIONING 2.	SVB12030
	MOVE SPACE TO P-R.	SVB12040
	WRITE P-R FROM VM-D AFTER POSITIONING 1.	SVB12050
	MOVE SPACE TO P-R VM-PR VM-D.	SVB12060
	ADD 1 TO ERRCNT.	SVB12070
	MOVE 0 TO SW.	SVB12080
	GO TO READ-VM.	SVB12090
VM-COUNT.		SVB12100



IF DONGGW > MYRM COMPUTE DONGG = DONGG - 1.	SVB12110
IF MYEMON > DONGKW MOVE "OV" TO FB7	SVB12120
MOVE "ER" TO FB8	SVB12130
GO TO VM-ERROR.	SVB12140
IF FYEMON > DONGKW MOVE "OV" TO FB9	SVB12150
MOVE "ER" TO FC1	SVB12160
GO TO VM-ERROR.	SVB12170
IF RMMON > RDONGW COMPUTE MAGG = DONGG - MYR - 1	SVB12180
ELSE COMPUTE MAGG = DONGG - MYR.	SVB12190
IF RFMON > RDONGW COMPUTE FAGG = DONGG - FYR - 1	SVB12200
ELSE COMPUTE FAGG = DONGG - FYR.	SVB12210
IF DONGG > 78 IF MAGG < 14 MOVE "##" TO FD3 GO TO VM-ERROR.	SVB12220
IF DONGG > 78 IF FAGG < 14 MOVE "##" TO FD4 GO TO VM-ERROR.	SVB12230
ADD 1 TO OCNT.	SVB12240
REWRITE DM-R.	SVB12250
GO TO READ-VM.	SVB12260
VM-END.	SVB12270
GO TO END-RTN.	SVB12280
VI-ED.	SVB12290
READ D-F AT END GO TO END-RTN.	SVB12300
ADD 1 TO ECNT.	SVB12310
MOVE JMN TO MBG.	SVB12320
MOVE JHDO TO DOBG.	SVB12330
MOVE JHGUN TO GUNBG.	SVB12340
GO TO VI-BI.	SVB12350
READ-VI.	SVB12360
READ D-F AT END GO TO VI-END.	SVB12370
ADD 1 TO ECNT.	SVB12380
VI-BI.	SVB12390
MOVE BUHO TO JBU.	SVB12400
MOVE YEARV TO JYR.	SVB12410
MOVE JHDO TO DOB.	SVB12420
MOVE JHGUN TO GUNB.	SVB12430
IF NOT JMNCD	SVB12440
MOVE "***" TO GA3	SVB12450
MOVE 0 TO SW.	SVB12460
PERFORM DO-RTN THRU DO-EX.	SVB12470
IF SWD = 1	SVB12480
MOVE "***" TO GA4	SVB12490
MOVE 0 TO SWD.	SVB12500
IF SWG = 1	SVB12510
MOVE "***" TO GA5	SVB12520
MOVE 0 TO SWG.	SVB12530
MOVE JBDO TO DOB.	SVB12540
MOVE JBGUN TO GUNB.	SVB12550
PERFORM DO-RTN THRU DO-EX.	SVB12560
IF SWD = 1	SVB12570
MOVE "***" TO GAB	SVB12580
MOVE 0 TO SWD.	SVB12590
IF SWG = 1	SVB12600
MOVE "***" TO GA9	SVB12610
MOVE 0 TO SWG.	SVB12620
MOVE JMDO TO DOB.	SVB12630
MOVE JMGUN TO GUNB.	SVB12640
PERFORM DO-RTN THRU DO-EX.	SVB12650

IF SWD = 1		SVB12660
	MOVE "***" TO GB1	SVB12670
	MOVE 0 TO SWD.	SVB12680
IF SWG = 1		SVB12690
	MOVE "***" TO GB2	SVB12700
	MOVE 0 TO SWG.	SVB12710
MOVE JFDO TO DOB.		SVB12720
MOVE JFGUN TO GUNB.		SVB12730
PERFORM DO-RTN THRU DO-EX.		SVB12740
IF SWD = 1		SVB12750
	MOVE "***" TO GB3	SVB12760
	MOVE 0 TO SWD.	SVB12770
IF SWG = 1		SVB12780
	MOVE "***" TO GB4	SVB12790
	MOVE 0 TO SWG.	SVB12800
IF MSYR NOT NUMERIC		SVB12810
	MOVE "***" TO GB5	SVB12820
	MOVE 1 TO SW.	SVB12830
IF NOT MSWOLCD		SVB12840
	MOVE "***" TO GB6	SVB12850
	MOVE 1 TO SW.	SVB12860
IF FSYR NOT NUMERIC		SVB12870
	MOVE "***" TO GB7	SVB12880
	MOVE 1 TO SW.	SVB12890
IF NOT FSWOLCD		SVB12900
	MOVE "***" TO GB8	SVB12910
	MOVE 1 TO SW.	SVB12920
IF DONGN NOT NUMERIC		SVB12930
	MOVE "***" TO GB9	SVB12940
	MOVE 1 TO SW.	SVB12950
IF NOT GUEWCD		SVB12960
	MOVE "***" TO GC1	SVB12970
	MOVE 1 TO SW.	SVB12980
IF DIVY NOT NUMERIC		SVB12990
	MOVE "***" TO GC2	SVB13000
	MOVE 1 TO SW.	SVB13010
IF NOT DIVWCD		SVB13020
	MOVE "***" TO GC3	SVB13030
	MOVE 1 TO SW.	SVB13040
IF NOT SUNGHCD		SVB13050
	MOVE "*" TO GC4	SVB13060
	MOVE 1 TO SW.	SVB13070
IF NOT DIVRESCD		SVB13080
	MOVE "*" TO GC5	SVB13090
	MOVE 1 TO SW.	SVB13100
IF NOT HEDCD		SVB13110
	MOVE "*" TO GC6	SVB13120
	MOVE 1 TO SW.	SVB13130
IF NOT WEDCD		SVB13140
	MOVE "*" TO GC7	SVB13150
	MOVE 1 TO SW.	SVB13160
IF SW = 0 GO TO VI-COUNT.		SVB13170
VI-ERROR.		SVB13180
MOVE JBU TO CA1.		SVB13190
MOVE JYR TO CA2.		SVB13200

MOVE JMN	TO CA3.	SVB13210
MOVE JHDO	TO CA4.	SVB13220
MOVE JHGUN	TO CA5.	SVB13230
MOVE JSOK	TO CA6.	SVB13240
MOVE JIL	TO CA7.	SVB13250
MOVE DIVOR-B	TO VIIII.	SVB13260
WRITE P-R	FROM COLDPY AFTER POSITIONING 2.	SVB13270
MOVE SPACE	TO P-R.	SVB13280
WRITE P-R	FROM VI-PR AFTER POSITIONING 2.	SVB13290
MOVE SPACE	TO P-R.	SVB13300
WRITE P-R	FROM VI-D AFTER POSITIONING 1.	SVB13310
MOVE SPACE	TO P-R VI-PR VI-D.	SVB13320
ADD 1	TO ERRCNT.	SVB13330
MOVE 0	TO SW.	SVB13340
GO TO	READ-VI.	SVB13350
VI-COUNT.		SVB13360
IF MSYEWOL >	DIVYWW MOVE "OV" TO GB5	SVB13370
	MOVE "ER" TO GB6	SVB13380
	GO TO VI-ERROR.	SVB13390
IF FSYEWOL >	DIVYWW MOVE "OV" TO GB7	SVB13400
	MOVE "ER" TO GB8	SVB13410
	GO TO VI-ERROR.	SVB13420
IF DONGKNW >	DIVYWW MOVE "OV" TO GB9	SVB13430
	MOVE "ER" TO GC1	SVB13440
	GO TO VI-ERROR.	SVB13450
IF RMSWOL <	RDIVW COMPUTE HAG = DIVY - MSYR - 1	SVB13460
ELSE	COMPUTE HAG = DIVY - MSYR.	SVB13470
IF HAG <	16 MOVE 16 TO HAG.	SVB13480
IF RFSWOL <	RDIVW COMPUTE EAG = DIVY - FSYR - 1	SVB13490
ELSE	COMPUTE HAG = DIVY - FSYR.	SVB13500
IF EAG <	18 MOVE 18 TO EAG.	SVB13510
IF RGUEW <	DIVY COMPUTE DONGK = DIVY - DONGN - 1	SVB13520
ELSE	COMPUTE DONGK = DIVY - DONGN.	SVB13530
IF DONGK >	HAG COMPUTE DONGK = HAG - 16.	SVB13540
	REWRITE DI-R.	SVB13550
	ADD 1 TO OCNT.	SVB13560
	GO TO READ-VI.	SVB13570
VI-END.		SVB13580
GO TO	END-RTN.	SVB13590
DO-RTN.		SVB13600
PERFORM	DO-L VARYING I FROM 1 BY 1 UNTIL I > DON(1).	SVB13610
	MOVE 1 TO SWD SW.	SVB13620
	GO TO DO-EX.	SVB13630
DO-L.		SVB13640
IF DOB =	GAT (I) GO TO GUN-SER.	SVB13650
GUN-SER.		SVB13660
IF I >	DON(2) COMPUTE M = DON(2) + 1	SVB13670
	MOVE M TO I.	SVB13680
GO TO	SEOUL BUSAN DAEGOO INCHEON KKEE KWON CHBUK CHNAM	SVB13690
	JEONBUK JEONNAM KBUK KNAM JEJU EXTRA DEPENDING ON I.	SVB13700
SEOUL.		SVB13710
PERFORM	SE-L VARYING J FROM 1 BY 1 UNTIL J > DON(3).	SVB13720
	MOVE 1 TO SWG SW.	SVB13730
	GO TO DO-EX.	SVB13740
SE-L.		SVB13750

IF GUNB = SE(J) GO TO DO-EX.	SVB13760
BUSAN.	SVB13770
PERFORM BU-L VARYING J FROM 1 BY 1 UNTIL J > DON(4).	SVB13780
MOVE 1 TO SWG SW.	SVB13790
GO TO DO-EX.	SVB13800
BU-L.	SVB13810
IF GUNB = BS(J) GO TO DO-EX.	SVB13820
DAEGOO.	SVB13830
PERFORM DA-L VARYING J FROM 1 BY 1 UNTIL J > DON(5).	SVB13840
MOVE 1 TO SWG SW.	SVB13850
GO TO DO-EX.	SVB13860
DA-L.	SVB13870
IF GUNB = DG(J) GO TO DO-EX.	SVB13880
INCHEON.	SVB13890
PERFORM IN-L VARYING J FROM 1 BY 1 UNTIL J > DON(6).	SVB13900
MOVE 1 TO SWG SW.	SVB13910
GO TO DO-EX.	SVB13920
IN-L.	SVB13930
IF GUNB = IC(J) GO TO DO-EX.	SVB13940
KKEE.	SVB13950
PERFORM KK-L VARYING J FROM 1 BY 1 UNTIL J > DON(7).	SVB13960
MOVE 1 TO SWG SW.	SVB13970
GO TO DO-EX.	SVB13980
KK-L. IF GUNB = KK(J) GO TO DO-EX.	SVB13990
KWON.	SVB14000
PERFORM KW-L VARYING J FROM 1 BY 1 UNTIL J > DON(8).	SVB14010
MOVE 1 TO SWG SW.	SVB14020
GO TO DO-EX.	SVB14030
KW-L. IF GUNB = KW(J) GO TO DO-EX.	SVB14040
CHBUK.	SVB14050
PERFORM CB-L VARYING J FROM 1 BY 1 UNTIL J > DON(9).	SVB14060
MOVE 1 TO SWG SW.	SVB14070
GO TO DO-EX.	SVB14080
CB-L. IF GUNB = CB(J) GO TO DO-EX.	SVB14090
CHNAM.	SVB14100
PERFORM CN-L VARYING J FROM 1 BY 1 UNTIL J > DON(10).	SVB14110
MOVE 1 TO SWG SW.	SVB14120
GO TO DO-EX.	SVB14130
CN-L. IF GUNB = CN(J) GO TO DO-EX.	SVB14140
JEONBUK.	SVB14150
PERFORM JB-L VARYING J FROM 1 BY 1 UNTIL J > DON(11).	SVB14160
MOVE 1 TO SWG SW.	SVB14170
GO TO DO-EX.	SVB14180
JB-L. IF GUNB = JB(J) GO TO DO-EX.	SVB14190
JEONNAM.	SVB14200
PERFORM JN-L VARYING J FROM 1 BY 1 UNTIL J > DON(12).	SVB14210
MOVE 1 TO SWG SW.	SVB14220
GO TO DO-EX.	SVB14230
JN-L. IF GUNB = JN(J) GO TO DO-EX.	SVB14240
KBUK.	SVB14250
PERFORM KB-L VARYING J FROM 1 BY 1 UNTIL J > DON(13).	SVB14260
MOVE 1 TO SWG SW.	SVB14270
GO TO DO-EX.	SVB14280
KB-L. IF GUNB = KB(J) GO TO DO-EX.	SVB14290
KNAM.	SVB14300

```

PERFORM KN-L          VARYING J FROM 1 BY 1 UNTIL J > DON(14).SVB14310
MOVE 1 TO SWG SW.    SVB14320
GO TO DO-EX.         SVB14330
KN-L. IF GUNB = KN(J) GO TO DO-EX.    SVB14340
JEJU.                SVB14350
PERFORM JJ-L          VARYING J FROM 1 BY 1 UNTIL J > DON(15).SVB14360
MOVE 1 TO SWG SW.    SVB14370
GO TO DO-EX.         SVB14380
JJ-L. IF GUNB = JJ(J) GO TO DO-EX.    SVB14390
EXTRA.               SVB14400
IF GUNB NOT = "00" MOVE 1 TO SWG SW.  SVB14410
DO-EX. EXIT.         SVB14420
END-RTN.             SVB14430
DISPLAY " ECNT = " ECNT.              SVB14440
DISPLAY " OCNT = " OCNT.              SVB14450
DISPLAY " ERRCNT = " ERRCNT.          SVB14460
CLOSE D-F P-F.       SVB14470
STOP RUN.            SVB14480
DISPLAY " ERRCNT = " ERRCNT.          SVB14490
CLOSE D-F P-F.       SVB14500
STOP RUN.            SVB14510
/*                  SVB14520
//GO.SYSIN DD *     SVB14530
112122233132333435363738390102030405 SVB14540
83                  SVB14550
1                      SVB14560
181317100604292113191831312704       SVB14570
1112131415161718192021222324252627   SVB14580
11121314151617181920                 SVB14590
111213141516                         SVB14600
11121314                               SVB14610
1112131415161718313233343536373839404142434445464748495051 SVB14620
111213141516313233343536373839404142434445 SVB14630
11121331323334353637383940           SVB14640
01021131323334353637383940414243444500 SVB14650
111213141531323334353637383940414243   SVB14660
01020311121314313233343536373839404142434445464748495051525300 SVB14670
11121314151617313233343536373839404142434445464748495051525354 SVB14680
111213141516171831323334353637383940414243444546474849 SVB14690
11123132                               SVB14700
1                      SVB14710
/*                  SVB14720
//GO.DISKIN DD DSN=SVBWED14(INDEX),DISP=OLD,UNIT=DISK,VOL=SER=WK2222 SVB14730
//          DD DSN=SVBWED14(PRIME),DISP=OLD,UNIT=DISK,VOL=SER=WK2222 SVB14740
//          DD DSN=SVBWED14(OVFLOW),DISP=OLD,UNIT=DISK,VOL=SER=WK2222 SVB14750
//GO.PROUT DD SYSOUT=A SVB14760
//SYSOUT DD SYSOUT=A SVB14770
// SVB14780

```

```

//SVBWPB14 JOB TYPRUN=HOLD
//*   BDMI   1 - 6 WOL  UPDATING   ***
//UPDATE EXEC COBUCLG,FARM='NOSOURCE'
//COB.SYSIN DD *
  ID DIVISION.
  PROGRAM-ID.   EDMICR.
  ENVIRONMENT  DIVISION.
  INPUT-OUTPUT SECTION.
  FILE-CONTROL.
    SELECT T-F ASSIGN UT-S-TAPEIN.
    SELECT DO-F ASSIGN DA-I-DISKOUT
      ACCESS MODE RANDOM
      NOMINAL KEY NOMI-KEY
      RECORD KEY DBIGYO.
    SELECT P-F ASSIGN UT-S-PROUT.
  DATA DIVISION.
  FILE SECTION.
  FD T-F          LABEL RECORD OMITTED.
  01 T-R.
    03 BIGYO     PIC X(16).
    03 TIGYO     REDEFINES BIGYO PIC 9(16).
    03 TT        OCCURS 61 PIC X.
    03 TEL       PIC XX.
    03 TF        PIC X.
  FD DO-F        BLOCK CONTAINS 40 RECORDS
  LABEL RECORD STANDARD.
  01 DO-R.
    03 DBIGYO    PIC X(16).
    03 DBG       REDEFINES DBIGYO.
    05 F         PIC X(13).
    05 IL        PIC X(3).
    03 DIGYO     REDEFINES DBIGYO PIC 9(16).
    03 DD        OCCURS 61 PIC X.
    03 DEL       PIC XX.
    03 DF        PIC X.
  FD P-F        LABEL RECORD OMITTED.
  01 P-R.
    03 F         PIC X.
    03 P1        PIC X(131).
  WORKING-STORAGE SECTION.
  77 ECNT PIC 9(7) VALUE 0.
  77 OCNT PIC 9(7) VALUE 0.
  77 ADD-CNT PIC 9(7) VALUE 0.
  77 CDR-CNT PIC 9(7) VALUE 0.
  77 DEL-CNT PIC 9(7) VALUE 0.
  77 NOMI-KEY PIC X(16).
  77 I        PIC 999 VALUE 0.
  77 J        PIC 999 VALUE 0.
  PROCEDURE DIVISION.
  OPEN INPUT T-F.
  OPEN OUTPUT P-F.
  OPEN I-O DO-F.
  MOVE SPACE TO P-R.
  WRITE P-R AFTER POSITIONING 0.
  MOVE SPACE TO P-R.

```

SVB00010  
SVB00020  
SVB00030  
SVB00040  
SVB00050  
SVB00060  
SVB00070  
SVB00080  
SVB00090  
SVB00100  
SVB00110  
SVB00120  
SVB00130  
SVB00140  
SVB00150  
SVB00160  
SVB00170  
SVB00180  
SVB00190  
SVB00200  
SVB00210  
SVB00220  
SVB00230  
SVB00240  
SVB00250  
SVB00260  
SVB00270  
SVB00280  
SVB00290  
SVB00300  
SVB00310  
SVB00320  
SVB00330  
SVB00340  
SVB00350  
SVB00360  
SVB00370  
SVB00380  
SVB00390  
SVB00400  
SVB00410  
SVB00420  
SVB00430  
SVB00440  
SVB00450  
SVB00460  
SVB00470  
SVB00480  
SVB00490  
SVB00500  
SVB00510  
SVB00520  
SVB00530  
SVB00540  
SVB00550

```

READ-RTN.
  READ T-F AT END GO TO END-RTN.
  ADD 1 TO ECNT.
  MOVE BIGYO TO NOMI-KEY.
  IF TF = 1 PERFORM COR-RTN THRU COR-EX
  ELSE IF TF = 3 PERFORM ADD-RTN THRU ADD-EX
  ELSE IF TF = 2 PERFORM DEL-RTN THRU DEL-EX
  ELSE DISPLAY "CODEER " T-R.
  GO TO READ-RTN.
COR-RTN.
  READ DO-F INVALID KEY GO TO KEY-ERROR.
  PERFORM BB THRU EX VARYING I FROM 1 BY 1 UNTIL I > 61.
  MOVE T-R TO P1.
  WRITE P-R AFTER POSITIONING 2.
  MOVE SPACE TO P-R.
  MOVE DO-R TO P1.
  WRITE P-R AFTER POSITIONING 1.
  MOVE SPACE TO P-R.
  REWRITE DO-R INVALID KEY DISPLAY " KEY =" BIGYO.
  ADD 1 TO COR-CNT.
COR-EX. EXIT.
BB. IF TT(I) NOT = " " MOVE TT(I) TO DD(I).
EX. EXIT.
ADD-RTN.
  ADD 1 TO ADD-CNT.
  DISPLAY "ADD-R " T-R.
  MOVE SPACES TO TEL TF.
  WRITE DO-R FROM T-R INVALID KEY DISPLAY " KEY =" BIGYO.
ADD-EX. EXIT.
DEL-RTN.
  ADD 1 TO DEL-CNT.
  DISPLAY "DEL-R " T-R.
  READ DO-F INVALID KEY GO TO KEY-ERROR.
  MOVE "**" TO DEL.
  REWRITE DO-R INVALID KEY DISPLAY " KEY =" BIGYO.
DEL-EX. EXIT.
KEY-ERROR.
  DISPLAY " KEY = " BIGYO.
  GO TO READ-RTN.
END-RTN.
  DISPLAY " ECNT = " ECNT.
  DISPLAY " OCNT = " OCNT.
  DISPLAY " ADDCNT = " ADD-CNT.
  DISPLAY " CORCNT = " COR-CNT.
  DISPLAY " DELCNT = " DEL-CNT.
  CLOSE T-F DO-F P-F.
  STOP RUN.
/*
//GO.TAPEIN DD DSN=SNOW14,UNIT=TAPE,DISP=(OLD,KEEP),VOL=SER=CHOIN1,
// DCB=EROPT=SKP,LABEL=(,NL)
//GO.DISKOUT DD DSN=SVBWED14(INDEX),UNIT=DISK,DISP=OLD,VOL=SER=WK2222
// DD DSN=SVBWED14(PRIME),UNIT=DISK,DISP=OLD,VOL=SER=WK2222
// DD DSN=SVBWED14(OVFLOW),UNIT=DISK,DISP=OLD,VOL=SER=WK2222
//GO.PROUT DD SYSOUT=A
//SYSOUT DD SYSOUT=A

```

FILE: SVRWPB14 UPDATE A1 VM/SP RELEASE 2.1 EXPRESS PUT8302+ SLU203

//

SVB01110



```

//SVBWAC14 JOB CLASS=V, TYPRUN=HOLD
//CREATION EXEC COBUCLG
//COB.SYSIN DD *
    ID DIVISION.
    PROGRAM-ID.    RVISAM.
    ENVIRONMENT DIVISION.
    INPUT-OUTPUT SECTION.
    FILE-CONTROL.
        SELECT DD-F ASSIGN DA-I-DISKOUT
            ACCESS MODE RANDOM
            NOMINAL KEY NOMI-KEY
            RECORD KEY VB-KEY.

    DATA DIVISION.
    FILE SECTION.
    FD  DD-F          BLOCK CONTAINS 40 RECORDS
        LABEL RECORD STANDARD.

    01  DD-R.
        03  VB-KEY PIC X(16).
        03  DD OCCURS 32 PIC XX.
    WORKING-STORAGE SECTION.
    77  NOMI-KEY PIC X(16).
    01  ACPT PIC X(16).
    PROCEDURE DIVISION.
        ACCEPT ACPT.
        OPEN I-O DD-F.

*1
        MOVE ACPT TO NOMI-KEY.
        READ DD-F INVALID KEY GO TO END-RTN.
        MOVE 60      TO DD(13).
        MOVE 63      TO DD(14).
        REWRITE DD-R INVALID KEY DISPLAY " KEY " VB-KEY.

*
        CLOSE DD-F.
        STOP RUN.
    END-RTN.
        DISPLAY " KEY " VB-KEY.
        CLOSE DD-F. STOP RUN.

/*
//GO.SYSIN DD *
1830936520302093
/*
//GO.DISKOUT DD DSN=SVBWED14(INDEX), UNIT=DISK, DISP=(OLD,KEEP),
//          VOL=SER=WK2222
//          DD DSN=SVBWED14(PRIME), UNIT=DISK, DISP=(OLD,KEEP),
//          VOL=SER=WK2222
//          DD DSN=SVBWED14(OVFLOW), UNIT=DISK, DISP=(OLD,KEEP),
//          VOL=SER=WK2222
//SYSOUT DD SYSOUT=A
//
SVB00010
SVB00020
SVB00030
SVB00040
SVB00050
SVB00060
SVB00070
SVB00080
SVB00090
SVB00100
SVB00110
SVB00120
SVB00130
SVB00140
SVB00150
SVB00160
SVB00170
SVB00180
SVB00190
SVB00200
SVB00210
SVB00220
SVB00230
SVB00240
SVB00250
SVB00260
SVB00270
SVB00280
SVB00290
SVB00300
SVB00310
SVB00320
SVB00330
SVB00340
SVB00350
SVB00360
SVB00370
SVB00380
SVB00390
SVB00400
SVB00410
SVB00420
SVB00430
SVB00440
SVB00450
SVB00460
SVB00470
SVB00480
SVB00490

```

```

//SVBWBK14 JOB CLASS=V, TYPRUN=HOLD
//*      BDMI WOL DISK TO TAPE BACKUP   ***
//BACKUP EXEC COBUCLG
//COB.SYSIN DD *
    ID DIVISION.
    PROGRAM-ID.    EDMICR.
    ENVIRONMENT DIVISION.
    INPUT-OUTPUT SECTION.
    FILE-CONTROL.
        SELECT DO-F ASSIGN DA-I-DISKIN
                ACCESS MODE SEQUENTIAL
                RECORD KEY VBDMDI.
        SELECT T-F ASSIGN UT-S-TAPEOUT.
    DATA DIVISION.
    FILE SECTION.
    FD DO-F      BLOCK CONTAINS 40 RECORDS
                LABEL RECORD STANDARD.
01  DO-R.
    03 VBDMDI   PIC X(16).
    03 RVB REDEFINES VBDMDI.
        05 F      PIC X(5).
        05 HJ     PIC X(4).
        05 F      PIC X(7).
    03 F      PIC X(61).
    03 DEL    PIC XX.
    03 F      PIC X.
    FD T-F      BLOCK CONTAINS 40 RECORDS
                LABEL RECORD STANDARD.
01  T-R      PIC X(80).
WORKING-STORAGE SECTION.
77 ECNT PIC 9(7) VALUE 0.
77 DCNT PIC 9(7) VALUE 0.
77 QCNT PIC 9(7) VALUE 0.
77 QCYO PIC 9(16) VALUE 0.
PROCEDURE DIVISION.
    OPEN INPUT DO-F OUTPUT T-F.
RD-RTN.
    READ DO-F AT END GO TO END-RTN.
    ADD 1 TO ECNT.
    IF DEL = " **" ADD 1 TO DCNT GO TO RD-RTN.
    EXAMINE DO-R REPLACING ALL SPACE BY 0.
    IF HJ = "3447" MOVE "3833" TO HJ.
    WRITE T-R FROM DO-R.
    MOVE SPACES TO T-R.
    ADD 1 TO QCNT.
    GO TO RD-RTN.
END-RTN.
    DISPLAY " INDISK COUNT = " ECNT.
    DISPLAY " DELETE COUNT = " DCNT.
    DISPLAY " OUTTAPECOUNT = " QCNT.
    DISPLAY " QUEST COUNT = " QCYO.
    CLOSE DO-F T-F.
    STOP RUN.

```

SVB00010  
SVB00020  
SVB00030  
SVB00040  
SVB00050  
SVB00060  
SVB00070  
SVB00080  
SVB00090  
SVB00100  
SVB00110  
SVB00120  
SVB00130  
SVB00140  
SVB00150  
SVB00160  
SVB00170  
SVB00180  
SVB00190  
SVB00200  
SVB00210  
SVB00220  
SVB00230  
SVB00240  
SVB00250  
SVB00260  
SVB00270  
SVB00280  
SVB00290  
SVB00300  
SVB00310  
SVB00320  
SVB00330  
SVB00340  
SVB00350  
SVB00360  
SVB00370  
SVB00380  
SVB00390  
SVB00400  
SVB00410  
SVB00420  
SVB00430  
SVB00440  
SVB00450  
SVB00460  
SVB00470  
SVB00480  
SVB00490  
SVB00500  
SVB00510  
SVB00520  
SVB00530  
SVB00540  
SVB00550

FILE: SVBWBK14 BACKUPB A1 VM/SP RELEASE 2.1 EXPRESS PUT8302+ SLU203

```
//GO.TAPEOUT DD DSN=SNOW14,UNIT=TAPE,DISP=NEW,VOL=SER=VBDMI1, SVB00560
//          DCE=EROPT=SKP,LABEL=(2,BLP) SVB00570
//GO.DISKIN DD DSN=SVBWED14(INDEX),DISP=OLD,UNIT=DISK,VOL=SER=WK2222 SVB00580
//          DD DSN=SVBWED14(PRIME),DISP=OLD,UNIT=DISK,VOL=SER=WK2222 SVB00590
//          DD DSN=SVBWED14(OVFLOW),DISP=OLD,UNIT=DISK,VOL=SER=WK2222 SVB00600
//GO.PROUT DD SYSOUT=A SVB00610
//SYSOUT DD SYSOUT=A SVB00620
// SVB00630
```

```

//SVBWMG14 JOB CLASS=A, TYPRUN=HOLD SVB00010
//MERGE EXEC COBUCLG, PARM='CLIST, STATE, FLOW, NUM' SVB00020
//COB.SYSIN DD * SVB00030
    ID DIVISION. SVB00040
    PROGRAM-ID. MG14. SVB00050
    ENVIRONMENT DIVISION. SVB00060
    INPUT-OUTPUT SECTION. SVB00070
    FILE-CONTROL. SVB00080
        SELECT T-F ASSIGN UT-S-TAPEIN. SVB00090
        SELECT O-F ASSIGN UT-S-TAPEOUT. SVB00100
    DATA DIVISION. SVB00110
    FILE SECTION. SVB00120
    FD T-F LABEL RECORD STANDARD SVB00130
        BLOCK CONTAINS 40 RECORDS. SVB00140
    01 T-R PIC X(80). SVB00150
    FD O-F LABEL RECORD STANDARD SVB00160
        BLOCK CONTAINS 40 RECORDS. SVB00170
    01 O-R PIC X(80). SVB00180
    WORKING-STORAGE SECTION. SVB00190
    77 T-CNT PIC 9(10) VALUE ZEROS. SVB00200
    77 O-CNT PIC 9(10) VALUE ZEROS. SVB00210
    PROCEDURE DIVISION. SVB00220
    SIJAK. SVB00230
        OPEN INPUT T-F OUTPUT O-F. SVB00240
    READ-T. SVB00250
        READ T-F AT END GO TO END-R. SVB00260
        ADD 1 TO T-CNT. SVB00270
        WRITE O-R FROM T-R. SVB00280
        ADD 1 TO O-CNT. SVB00290
        GO TO READ-T. SVB00300
    END-R. SVB00310
        CLOSE T-F O-F. SVB00320
        DISPLAY " INPUT COUNT " T-CNT. SVB00330
        DISPLAY "OUTPUT COUNT " O-CNT. SVB00340
        STOP RUN. SVB00350
/* SVB00360
//GO.STEPLIB DD DSN=SYS1.COBLIB, DISP=SHR SVB00370
//GO.TAPEIN DD DSN=HA, UNIT=TAPE, DISP=(OLD, KEEP), LABEL=(2, BLP), SVB00380
// DCB=(LRECL=80, BLKSIZE=3200, RECFM=FB, EROPT=SKP), VOL=SER=000001 SVB00390
// DD DSN=HB, UNIT=AFF=TAPEIN, LABEL=(2, BLP), VOL=SER=000002 SVB00400
// DD DSN=HB, UNIT=AFF=TAPEIN, LABEL=(2, BLP), VOL=SER=000003 SVB00410
// DD DSN=HB, UNIT=AFF=TAPEIN, LABEL=(2, BLP), VOL=SER=000004 SVB00420
// DD DSN=HB, UNIT=AFF=TAPEIN, LABEL=(2, BLP), VOL=SER=000005 SVB00430
// DD DSN=HB, UNIT=AFF=TAPEIN, LABEL=(2, BLP), VOL=SER=000006 SVB00440
// DD DSN=HB, UNIT=AFF=TAPEIN, LABEL=(2, BLP), VOL=SER=000007 SVB00450
// DD DSN=HB, UNIT=AFF=TAPEIN, LABEL=(2, BLP), VOL=SER=000008 SVB00460
// DD DSN=HB, UNIT=AFF=TAPEIN, LABEL=(2, BLP), VOL=SER=000009 SVB00470
// DD DSN=HB, UNIT=AFF=TAPEIN, LABEL=(2, BLP), VOL=SER=000010 SVB00480
// DD DSN=HB, UNIT=AFF=TAPEIN, LABEL=(2, BLP), VOL=SER=000011 SVB00490
// DD DSN=HB, UNIT=AFF=TAPEIN, LABEL=(2, BLP), VOL=SER=000012 SVB00500
//GO.TAPEOUT DD DSN=HO, UNIT=TAPE, DISP=(NEW, KEEP), VOL=(, , 99), SVB00510
// SER=SCRATCH), DCB=(LRECL=80, BLKSIZE=3200, RECFM=FB, EROPT=SKP) SVB00520
//GO.SYSPRINT DD SYSOUT=A SVB00530
//GO.SYSDBOUT DD SYSOUT=A SVB00540
//GO.SYSOUT DD SYSOUT=A SVB00550

```

FILE: SVBWMG14 MERGE A1 VM/SP RELEASE 2.1 EXPRESS PUT8302+ SLU203

//

SVB00560

FILE: SVBUSB14 SORHJ A1 VM/SP RELEASE 2.1 EXPRESS PUT8302+ SLU203

```
//SVBUSB14 JOB TYPRUN=HOLD SVB00010
//SORT EXEC PGM=SORT,PARM='SIZE(MAX)' SVB00020
//SYSOUT DD SYSOUT=A SVB00030
//SYSPRINT DD SYSOUT=A SVB00040
//SORTIN DD UNIT=TAPE,VOL=SER=000001,LABEL=(2,BLP),DISP=(OLD,KEEP), SVB00050
// DSN=SRR14,DCB=(LRECL=80,RECFM=FB,BLKSIZE=3200,EROPT=SKP) SVB00060
//SORTOUT DD DISP=(NEW,KEEP),UNIT=TAPE,DSN=SRCMM14, SVB00070
// DCB=(LRECL=80,RECFM=FB,BLKSIZE=3200) SVB00080
//SORTWK01 DD UNIT=3350,SPACE=(CYL,(100)) SVB00090
//SORTWK02 DD UNIT=3350,SPACE=(CYL,(100)) SVB00100
SORT FIELDS=(1,9,A),FORMAT=CH SVB00110
/* SVB00120
// SVB00130
```

```

//SVBWTB14 JOB CLASS=V, TYPRUN=HOLD
//TABCNT EXEC PLIFCLG
//PLI.SYSIN DD *
* PROCESS GS, NEST, OPT(2);
  GOLD: PROC OPTIONS(MAIN);
    DCL TF FILE RECORD INPUT;
    DCL 1 TR,
        2 F1 CHAR(3),
        2 TMN PIC '99',
        2 TDO CHAR(2),
        2 TGN CHAR(2),
        2 F2 CHAR(71);
    DCL TOT(4) PIC '(10)9' INIT((4)0);
    DCL HD CHAR(6) INIT((6)' ');
    OPEN FILE(TF);
    ON ENDFILE(TF) GOTO LAST;
RD: READ FILE(TF) INTO(TR);
    TCNT = TCNT+1;
    IF TCNT = 1 THEN HD=TMN||TDO||TGN;
    IF SUBSTR(HD,1,2) ^= TMN THEN DO; CALL PMN;
        HD=TMN||TDO||TGN;
        END;
    IF SUBSTR(HD,1,4) ^= TMN||TDO THEN DO; CALL PDO;
        HD=TMN||TDO||TGN;
        END;
    IF HD ^= TMN||TDO||TGN THEN DO; CALL PGN;
        HD=TMN||TDO||TGN;
        END;

    TOT(4) = TOT(4) + 1;
    GOTO RD;
PGN: PROC;
    PUT SKIP(2) EDIT('83', SUBSTR(HD,1,2),
        SUBSTR(HD,3,2),
        SUBSTR(HD,5,2), TOT(4))
        (X(20), A, A, X(10), A, X(10), A, X(10), F(10));
    TOT(3)=TOT(3) + TOT(4);
    TOT(4)= 0;
END PGN;
PDO: PROC;
    CALL PGN;
    PUT SKIP(3) EDIT('* ', '83', SUBSTR(HD,1,2),
        SUBSTR(HD,3,2), TOT(3))
        (X(18), A, A, A, X(10), A, X(22), F(10));
    PUT SKIP(0) EDIT('* ', '83', SUBSTR(HD,1,2),
        SUBSTR(HD,3,2), TOT(3))
        (X(18), A, A, A, X(10), A, X(22), F(10));
    PUT SKIP(3);
    TOT(2)=TOT(2) + TOT(3);
    TOT(3)= 0;
END PDO;
PMN: PROC;
    CALL PDO;
    PUT SKIP(3) EDIT('**', '83', SUBSTR(HD,1,2), TOT(2))
        (X(18), A, A, A, X(34), F(10));
    PUT SKIP(0) EDIT('**', '83', SUBSTR(HD,1,2), TOT(2))

```

SVB00010  
SVB00020  
SVB00030  
SVB00040  
SVB00050  
SVB00060  
SVB00070  
SVB00080  
SVB00090  
SVB00100  
SVB00110  
SVB00120  
SVB00130  
SVB00140  
SVB00150  
SVB00160  
SVB00170  
SVB00180  
SVB00190  
SVB00200  
SVB00210  
SVB00220  
SVB00230  
SVB00240  
SVB00250  
SVB00260  
SVB00270  
SVB00280  
SVB00290  
SVB00300  
SVB00310  
SVB00320  
SVB00330  
SVB00340  
SVB00350  
SVB00360  
SVB00370  
SVB00380  
SVB00390  
SVB00400  
SVB00410  
SVB00420  
SVB00430  
SVB00440  
SVB00450  
SVB00460  
SVB00470  
SVB00480  
SVB00490  
SVB00500  
SVB00510  
SVB00520  
SVB00530  
SVB00540  
SVB00550

FILE: SVBWB14 HJCNT A1 VM/SF RELEASE 2.1 EXPRESS PUT8302+ SLU203

```

                                (X(18),A,A,A,X(34),F(10));
PUT SKIP(3);
TOT(1)=TOT(1) + TOT(2);
TOT(2)= 0;
PUT PAGE;
END PMN;
JUK: PROC;
CALL PMN;
PUT SKIP(6) EDIT('**', '83TT',TOT(1))
                                (X(18),A,A,X(34),F(10));
PUT SKIP(0) EDIT('**', '83TT',TOT(1))
                                (X(18),A,A,X(34),F(10));
END JUK;
LAST:
CALL JUK;
PUT SKIP(5) EDIT('INPUT COUNT = ',TCNT)(X(18),A,A);
CLOSE FILE(TF);
END GOLD;
/*
//GO.TF DD UNIT=TAPE,DSN=HH,VOL=SER=123321,LABEL=(2,BLP),
// DCB=(RECFM=FB,BLKSIZE=3200,LRECL=80),DISP=(OLD,KEEP)
//GO.SYSOUT DD SYSOUT=A
//GO.SYSPRINT DD SYSOUT=A
//
```

SVB00560  
SVB00570  
SVB00580  
SVB00590  
SVB00600  
SVB00610  
SVB00620  
SVB00630  
SVB00640  
SVB00650  
SVB00660  
SVB00670  
SVB00680  
SVB00690  
SVB00700  
SVB00710  
SVB00720  
SVB00730  
SVB00740  
SVB00750  
SVB00760  
SVB00770  
SVB00780  
SVB00790



```

//SVBWCBI4 JOB CLASS=A, TYPRUN=HOLD
//SELECT EXEC COBUCLG, PARM='CLIST, STATE, FLOW, NUM'
//COB.SYSIN DD *
    ID DIVISION.
    PROGRAM-ID. MG14.
    ENVIRONMENT DIVISION.
    INPUT-OUTPUT SECTION.
    FILE-CONTROL.
        SELECT T-F ASSIGN UT-S-TAPEIN.
        SELECT O-F ASSIGN UT-S-TAPEOUT.
    DATA DIVISION.
    FILE SECTION.
    FD T-F LABEL RECORD STANDARD
        BLOCK CONTAINS 40 RECORDS.
    01 T-R.
        02 TKEY          PIC X(13).
        02 F             PIC X(67).
    FD O-F LABEL RECORD STANDARD
        BLOCK CONTAINS 40 RECORDS.
    01 O-R             PIC X(80).
    WORKING-STORAGE SECTION.
    77 T-CNT          PIC 9(10) VALUE ZEROS.
    77 O-CNT          PIC 9(10) VALUE ZEROS.
    PROCEDURE DIVISION.
    SIJAK.
        OPEN INPUT T-F OUTPUT O-F.
    READ-T.
        READ T-F AT END GO TO END-R.
        ADD 1 TO T-CNT.
        IF TKEY = "1830211200604" MOVE "1830211180604" TO TKEY.
        IF TKEY = "1830211271211" MOVE "1830211211211" TO TKEY.
        IF TKEY = "1830321200402" MOVE "1830221200402" TO TKEY.
        IF TKEY = "1830436480505" MOVE "1830436380505" TO TKEY.
        IF TKEY = "1830638180201" MOVE "1830638380201" TO TKEY.
        IF TKEY = "2830338180101" MOVE "2830338380101" TO TKEY.
        IF TKEY = "2830336360101" MOVE "2830336530101" TO TKEY.
        IF TKEY = "4830537320201" MOVE "4830533320201" TO TKEY.
        IF TKEY = "4830623140301" MOVE "4830623130301" TO TKEY.
        IF TKEY = "4830638370301" MOVE "4830538370301" TO TKEY.
        IF TKEY = "4830638370302" MOVE "4830538370302" TO TKEY.
        WRITE O-R FROM T-R.
        ADD 1 TO O-CNT.
        GO TO READ-T.
    END-R.
        CLOSE T-F O-F.
        DISPLAY " INPUT COUNT " T-CNT.
        DISPLAY "OUTPUT COUNT " O-CNT.
        STOP RUN.
/*
//GO.TAPEIN DD DSN=HA, UNIT=TAPE, DISP=(OLD, KEEP), LABEL=(2, BLP),
// DCB=(LRECL=80, BLKSIZE=3200, RECFM=FB, EROPT=SKP), VOL=SER=000001
//GO.TAPEOUT DD DSN=HO, UNIT=TAPE, DISP=(NEW, KEEP), VOL=(, , 99,
// SER=SCRATCH), DCB=(LRECL=80, BLKSIZE=3200, RECFM=FB, EROPT=SKP)
//GO.SYSPRINT DD SYSOUT=A
//GO.SYSDBOUT DD SYSOUT=A

```

SVB00010  
SVB00020  
SVB00030  
SVB00040  
SVB00050  
SVB00060  
SVB00070  
SVB00080  
SVB00090  
SVB00100  
SVB00110  
SVB00120  
SVB00130  
SVB00140  
SVB00150  
SVB00160  
SVB00170  
SVB00180  
SVB00190  
SVB00200  
SVB00210  
SVB00220  
SVB00230  
SVB00240  
SVB00250  
SVB00260  
SVB00270  
SVB00280  
SVB00290  
SVB00300  
SVB00310  
SVB00320  
SVB00330  
SVB00340  
SVB00350  
SVB00360  
SVB00370  
SVB00380  
SVB00390  
SVB00400  
SVB00410  
SVB00420  
SVB00430  
SVB00440  
SVB00450  
SVB00460  
SVB00470  
SVB00480  
SVB00490  
SVB00500  
SVB00510  
SVB00520  
SVB00530  
SVB00540  
SVB00550

FILE: SVBMCB14 CORRECTI A1 VM/SP RELEASE 2.1 EXPRESS PUT8302+ SLU203

//GO.SYSOUT DD SYSOUT=A  
//

SVB00560  
SVB00570

```

//SVBWS114 JOB TYPRUN=HOLD
/**      BIRTH DEATH MARRIAGE DIVORCE  SUM 1  ***/
//      EXEC PLIFCLG
//PLI.SYSIN DD *
* PROCESS GS,NEST,OPT(TIME);
  SNOW:  PROC OPTIONS(MAIN) ;
         TAB1=0;
         OPEN FILE(TAPEINA) INPUT,
               FILE(TAPEOUT) OUTPUT,
               FILE(SYSPRINT) LINESIZE(132) PAGESIZE(66);
         ON ENDFILE(TAPEINA) END1_REC='YES';
         END1_REC='NO ' ;
/**      TAPE INPUT A  READ PART  **/
         READ FILE(TAPEINA) INTO(TA_R);
         M=F1;
         DO WHILE (END1_REC='NO ');
           CALL SET_RTN;
           READ FILE(TAPEINA) INTO(TA_R) ;
         END;
         CLOSE FILE(TAPEINA) ;
         CALL WRITE_RTN;
         CLOSE FILE(TAPEOUT);
DCL TAPEINA FILE RECORD INPUT;
DCL 1 TA_R,
    3 F1 CHAR(1),
    3 YR PIC '99',
    3 F2 CHAR(23),
    3 YEAR PIC '99',
    3 F3 CHAR(52);
DCL 1 TD_R DEF TA_R POS(1),
    3 DF1 CHAR(1),
    3 DYR PIC '99',
    3 DF2 CHAR(30),
    3 DYEAR PIC '99',
    3 DF3 CHAR(45);
DCL 1 TM_R DEF TA_R POS(1),
    3 MF1 CHAR(1),
    3 MYR PIC '99',
    3 MF2 CHAR(37),
    3 MYEAR PIC '99',
    3 MF3 CHAR(38);
DCL 1 TI_R DEF TA_R POS(1),
    3 IF1 CHAR(1),
    3 VYR PIC '99',
    3 IF2 CHAR(37),
    3 VYEAR PIC '99',
    3 IF3 CHAR(38);
DCL YRD(12) PIC '99' INIT(0,81,80,79,78,77,76,75,74,73,72,71);
DCL TAB1(12) FIXED DEC(9);
DCL TAPEOUT FILE RECORD OUTPUT;
DCL 1 TO_R,
    3 KU PIC '9',
    3 KU1 PIC '99',
    3 DD(20) PIC '(8)9';
DCL END1_REC CHAR(3);

```

10000010  
10000020  
10000030  
10000040  
10000050  
10000060  
10000070  
10000080  
10000090  
10000100  
10000110  
10000120  
10000130  
10000140  
10000150  
10000160  
10000170  
10000180  
10000190  
10000200  
10000210  
10000220  
10000230  
10000240  
10000250  
10000260  
10000270  
10000280  
10000290  
10000300  
10000310  
10000320  
10000330  
10000340  
10000350  
10000360  
10000370  
10000380  
10000390  
10000400  
10000410  
10000420  
10000430  
10000440  
10000450  
10000460  
10000470  
10000480  
10000490  
10000500  
10000510  
10000520  
10000530  
10000540  
10000550

```

/** SUB PROCEDURE **/
SET_RTN: PROC ;
    IF M=1 THEN DO ;
        IF YEAR < 71 THEN GO TO ADDSET ;
        DO N=2 BY 1 UNTIL(YEAR = YRD(N) | N > 12); END ;
        END ;
    ELSE
    IF M=2 THEN DO;
        IF DYEAR < 71 THEN GO TO ADDSET ;
        DO N=2 BY 1 UNTIL(DYEAR = YRD(N) | N > 12); END ;
        END ;
    ELSE
    IF M=4 THEN DO ;
        IF MYEAR < 71 THEN GO TO ADDSET ;
        DO N=2 BY 1 UNTIL(MYEAR = YRD(N) | N > 12); END ;
        END;
    ELSE DO ;
        IF VYEAR < 71 THEN GO TO ADDSET ;
        DO N=2 BY 1 UNTIL(VYEAR = YRD(N) | N > 12) ; END ;
        END ;
        IF N > 12 THEN GOTO ADDSET;
        TAB1(N) = TAB1(N) +1;
    ADDSET: TAB1(1) = TAB1(1) +1;
    END SET_RTN;
WRITE_RTN: PROC;
    KU=M;
    KU1 = 1;
    DO J=1 TO 12 ;
        DD(J) = TAB1(J);
    END ;
    WRITE FILE(TAPEOUT) FROM(TO_R);
END WRITE_RTN;
END SNOW;
/*
//GO.TAPEINA DD UNIT=TAPE,DSN=CHOI13,DISP=(OLD,KEEP),VOL=SER=CHOIN1,
// LABEL=(2,BLP),DCB=(LRECL=80,RECFM=FB,BLKSIZE=3200,EROPT=SKP)
//GO.TAPEOUT DD UNIT=TAPE,DSN=VITBS1,DISP=(NEW,KEEP),
// DCB=(LRECL=163,RECFM=FB,BLKSIZE=3260,EROPT=SKP)
//GO.SYSPRINT DD SYSOUT=A
//

```

10000560  
10000570  
10000580  
10000590  
10000600  
10000610  
10000620  
10000630  
10000640  
10000650  
10000660  
10000670  
10000680  
10000690  
10000700  
10000710  
10000720  
10000730  
10000740  
10000750  
10000760  
10000770  
10000780  
10000790  
10000800  
10000810  
10000830  
10000840  
10000850  
10000860  
10000870  
10000890  
10000900  
10000910  
10000920  
10000930  
10000940  
10000950  
10000960  
10000970

```

//SVBWT114 JOB TYPRUN=HOLD
//*      BIRTH DEATH MARRIAGE DIVORCE  SUM 1 ***
//      EXEC PLIFCLG
//PLI.SYSIN DD *
* PROCESS S,GS,NEST,OPT(TIME);
  CHOI:  PROC OPTIONS(MAIN) ;
         TABA=0;
         OPEN FILE(TAPEINA) INPUT,
              FILE(TAPEINB) INPUT,
              FILE(TAPEOUT) OUTPUT,
              FILE(SYSPRINT) LINESIZE(132) PAGESIZE(66);
         ON ENDFILE(TAPEINA) END1_REC='YES';
         ON ENDFILE(TAPEINB) END2_REC='YES';
         END1_REC='NO ';
         END2_REC='NO ';
/** TAPE INPUT A  READ PART **/
  READ FILE(TAPEINA) INTO(TA_R);
  DO WHILE (END1_REC='NO ');
    CALL SET_RTN;
    READ FILE(TAPEINA) INTO(TA_R) ;
  END;
  CLOSE FILE(TAPEINA) ;
/** TAPE INPUT B  READ PART **/
  READ FILE(TAPEINB) INTO(TB_R);
  DO WHILE(END2_REC='NO ');
    CALL SUM_RTN;
    READ FILE(TAPEINB) INTO(TB_R);
  END;
  CALL ADD_RTN;
  CALL WRITE_RTN;
  CLOSE FILE(TAPEINB),FILE(TAPEOUT);
DCL TAPEINA FILE RECORD INPUT;
DCL 1 TA_R,
    3 CU PIC '9',
    3 CU1 PIC '99',
    3 AA(20) PIC '(8)9';
DCL TAPEINB FILE RECORD INPUT;
DCL 1 TB_R,
    3 GU PIC '9',
    3 GU1 PIC '99',
    3 BB(20) PIC '(8)9';
DCL TAPEOUT FILE RECORD OUTPUT;
DCL 1 TO_R,
    3 KU PIC '9',
    3 KU1 PIC '99',
    3 DD(20) PIC '(8)9';
DCL TABA(4,12,12) FIXED DEC(9);
DCL END1_REC CHAR(3);
DCL END2_REC CHAR(3);
/** SUB PROCEDURE **/
SET_RTN: PROC ;
         IF CU = 1|CU=2 THEN H =CU;
         ELSE H =CU-1;
         IF CU1 = 2 THEN GO TO ENDSET;
         I = CU1;

```

SVB00010  
SVB00020  
SVB00030  
SVB00040  
SVB00050  
SVB00060  
SVB00070  
SVB00080  
SVB00090  
SVB00100  
SVB00110  
SVB00120  
SVB00130  
SVB00140  
SVB00150  
SVB00160  
SVB00170  
SVB00180  
SVB00190  
SVB00200  
SVB00210  
SVB00220  
SVB00230  
SVB00240  
SVB00250  
SVB00260  
SVB00270  
SVB00280  
SVB00290  
SVB00300  
SVB00310  
SVB00320  
SVB00330  
SVB00340  
SVB00350  
SVB00360  
SVB00370  
SVB00380  
SVB00390  
SVB00400  
SVB00410  
SVB00420  
SVB00430  
SVB00440  
SVB00450  
SVB00460  
SVB00470  
SVB00480  
SVB00490  
SVB00500  
SVB00510  
SVB00520  
SVB00530  
SVB00540  
SVB00550

```

DO J=1 TO 12;
    TABA(H,I,J) = AA(J);
END;
ENDSET: END SET_RTN;
SUM_RTN: PROC;
    IF GU = 1 | GU=2 THEN H =GU;
        ELSE H =GU-1;
    IF GU1 = 1 | GU1 = 12 THEN GO TO ENDSUM;
    I=GU1+1;
    TABA(H,I,1) = BB(1);
    DO J=2 TO 11;
        K=J+1;
        TABA(H,I,K) = BB(J);
    END;
ENDSUM: END SUM_RTN;
ADD_RTN: PROC;
    DO I =2 TO 12;
        TABA(*,I,*) = TABA(*,1,*) + TABA(*,I,*);
    END ;
END ADD_RTN;
WRITE_RTN: PROC;
    DO H = 1 TO 4;
        IF H = 1 | H = 2 THEN KU = H;
            ELSE KU = H +1;
        DO I=1 TO 12 ;
            KU1 = I;
            DO J=1 TO 12 ;
                DD(J) = TABA(H,I,J);
            END ;
            WRITE FILE(TAPEOUT) FROM(TO_R);
        END;
    END;
END WRITE_RTN;
END CHOI;
/*
//GO.TAPEINA DD UNIT=TAPE,DSN=CHOI13,DISP=(OLD,KEEP),VOL=SER=CH2237,
// LABEL=(2,BLF),DCB=(LRECL=163,RECFM=FB,BLKSIZE=3260,EROPT=SKP)
//GO.TAPEINB DD UNIT=TAPE,DSN=CHOISM,DISP=(OLD,KEEP),VOL=SER=CH4138,
// LABEL=(2,BLF),DCB=(LRECL=163,RECFM=FB,BLKSIZE=3260,EROPT=SKP)
//GO.TAPEOUT DD UNIT=TAPE,DSN=VITBS1,DISP=(NEW,KEEP),
// DCB=(LRECL=163,RECFM=FB,BLKSIZE=3260,EROPT=SKP)
//GO.SYSPRINT DD SYSOUT=A
//

```

```

//SVBWP114 JOB MSGLEVEL=(1,1),TYPRUN=HOLD
//TABED EXEC COBUCLG,PARM='NOSOURCE'
//COB.SYSIN DD *
ID DIVISION.
PROGRAM-ID.          TA1ED.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SPECIAL-NAMES.
        CO1 IS CHOI
        CSP IS DOUB.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
        SELECT T-F ASSIGN UT-S-TAPEIN.
        SELECT P-F ASSIGN UT-S-PROUT.
DATA DIVISION.
FILE SECTION.
FD T-F                BLOCK          CONTAINS    20 RECORDS
                       LABEL          RECORD     STANDARD.
01 T-R.
   03 GU PIC          9.
   03 GU1 PIC         99.
   03 AA OCCURS 20 PIC 9(8).
FD P-F                LABEL RECORD OMITTED.
01 P-R                PIC X(132).
WORKING-STORAGE SECTION.
77 I PIC 9(3) VALUE 0.
77 J PIC 9(3) VALUE 0.
77 K PIC 9(3) VALUE 0.
77 L PIC 9(3) VALUE 0.
77 M PIC 9(3) VALUE 2.
77 N PIC 9(3) VALUE 1.
77 CNT PIC 99 VALUE 0.
77 ECNT PIC 99 VALUE 0.
01 LIST1.
   03 F PIC X(38) VALUE SPACE.
   03 B1 PIC ZZZZ,ZZZ,ZZZ,ZZZ,ZZ9.
   03 BB OCCURS 4 PIC ZZ,ZZZ,ZZZ,ZZZ,ZZ9.
01 LIST2.
   03 F PIC X(3) VALUE SPACE.
   03 CC OCCURS 7 PIC ZZ,ZZZ,ZZZ,ZZZ,ZZ9.
PROCEDURE DIVISION.
OPEN INPUT T-F.
OPEN OUTPUT P-F.
** TAB 1 EDITING **
BUSAN.
MOVE SPACE TO P-R.
WRITE P-R AFTER ADVANCING CHOI.
MOVE SPACE TO P-R.
WRITE P-R AFTER ADVANCING 3.
MOVE SPACE TO P-R.
WRITE P-R AFTER ADVANCING 3.
MOVE SPACE TO P-R.
WRITE P-R AFTER ADVANCING 2.
MOVE SPACE TO P-R.
READ-RTN.

```

SVB00010  
SVB00020  
SVB00030  
SVB00040  
SVB00050  
SVB00060  
SVB00070  
SVB00080  
SVB00090  
SVB00100  
SVB00110  
SVB00120  
SVB00130  
SVB00140  
SVB00150  
SVB00160  
SVB00170  
SVB00180  
SVB00190  
SVB00200  
SVB00210  
SVB00220  
SVB00230  
SVB00240  
SVB00250  
SVB00260  
SVB00270  
SVB00280  
SVB00290  
SVB00300  
SVB00310  
SVB00320  
SVB00330  
SVB00340  
SVB00350  
SVB00360  
SVB00370  
SVB00380  
SVB00390  
SVB00400  
SVB00410  
SVB00420  
SVB00430  
SVB00440  
SVB00450  
SVB00460  
SVB00470  
SVB00480  
SVB00490  
SVB00500  
SVB00510  
SVB00520  
SVB00530  
SVB00540  
SVB00550

```

READ T-F AT END      GO TO END-RTN.                SVB00560
MOVE AA(1) TO B1.    VARYING I FROM 2 BY 1 UNTIL I > 5. SVB00570
PERFORM P2
IF GU1 = 1
    WRITE P-R AFTER ADVANCING 2                    SVB00600
    MOVE SPACE TO P-R                              SVB00610
    WRITE P-R FROM LIST1 AFTER ADVANCING 3         SVB00620
    WRITE P-R FROM LIST1 AFTER DOUB              SVB00630
    MOVE SPACE TO P-R                              SVB00640
ELSE
IF GU1 = 2
    WRITE P-R FROM LIST1 AFTER ADVANCING 3         SVB00660
    MOVE SPACE TO P-R                              SVB00670
ELSE
IF GU1 = 4 OR 6 OR 8 OR 10
    WRITE P-R FROM LIST1 AFTER ADVANCING 2         SVB00700
    MOVE SPACE TO P-R                              SVB00710
ELSE
WRITE P-R FROM LIST1 AFTER ADVANCING 1            SVB00730
MOVE SPACE TO P-R.                                SVB00740
GO TO READ-RTN.                                    SVB00760
P2. COMPUTE J = I - 1.                              SVB00770
MOVE AA (I) TO BB (J).                             SVB00780
END-RTN.                                            SVB00790
CLOSE T-F.                                          SVB00800
OPEN INPUT T-F.                                     SVB00810
PERFORM BUSAN.                                       SVB00820
RD-RTN.                                             SVB00830
READ T-F AT END      GO TO LAST-RTN.                SVB00840
PERFORM P3
IF GU1 = 1
    WRITE P-R AFTER ADVANCING 2                    SVB00870
    MOVE SPACE TO P-R                              SVB00880
    WRITE P-R FROM LIST2 AFTER ADVANCING 3         SVB00890
    WRITE P-R FROM LIST2 AFTER DOUB              SVB00900
    MOVE SPACE TO P-R                              SVB00910
ELSE
IF GU1 = 2
    WRITE P-R FROM LIST2 AFTER ADVANCING 3         SVB00930
    MOVE SPACE TO P-R                              SVB00940
ELSE
IF GU1 = 4 OR 6 OR 8 OR 10
    WRITE P-R FROM LIST2 AFTER ADVANCING 2         SVB00980
    MOVE SPACE TO P-R                              SVB00990
ELSE
WRITE P-R FROM LIST2 AFTER ADVANCING 1            SVB01000
MOVE SPACE TO P-R.                                SVB01010
GO TO RD-RTN.                                       SVB01020
P3. COMPUTE J = I - 5.                              SVB01030
MOVE AA (I) TO CC (J).                             SVB01040
LAST-RTN.                                           SVB01060
CLOSE T-F P-F.                                       SVB01070
STOP RUN.                                           SVB01080
/*
//GO.TAPEIN DD UNIT=TAPE,DSN=BDMITB,DISP=OLD,VOL=SER=BDMI04, SVB01090
SVB01100

```



FILE: SVBWP114 PRT1      A1   VM/SP RELEASE 2.1 EXPRESS PUT8302+ SLU203

//	LABEL=(2,BLP),DCB=(EROPT=SKP)	SVB01110
//SYSOUT	DD SYSOUT=A	SVB01120
//GO.PROUT	DD SYSOUT=A,COPIES=2	SVB01130
//SYSDROUT	DD SYSOUT=A	SVB01140
//		SVB01150

```

//SVBWT214 JOB MSGCLASS=A,CLASS=A,TYPRUN=HOLD
//TABED EXEC COBUCLG
//*      **      TABLE 2 BIRTH DEATH DIVORCE SUM *****
//COB.SYSIN DD *
    ID DIVISION.
    PROGRAM-ID.    TAB2.
    ENVIRONMENT DIVISION.
    INPUT-OUTPUT SECTION.
        SELECT T-F ASSIGN UT-S-TAPEIN.
        SELECT TA-F ASSIGN UT-S-TAPEOUT.
    DATA DIVISION.
    FILE SECTION.
    FD T-F          BLOCK CONTAINS 40 RECORDS
                   LABEL RECORD STANDARD.
01 T-R.
   03 BU           PIC 9.
   03 F           PIC X(19).
   03 DO          PIC XX.
   03 DOF         REDEFINES DO PIC 99.
   03 GUN         PIC XX.
   03 GUNF        REDEFINES GUN PIC 99.
   03 F           PIC XX.
   03 YR         PIC XX.
   03 F           PIC X(52).
FD TA-F          BLOCK CONTAINS 20 RECORDS
                   LABEL RECORD STANDARD.
01 TA-R.
   03 GU1         PIC 999.
   03 GU2         PIC 99.
   03 GU          PIC 99.
   03 AA OCCURS 11 PIC 9(8).
WORKING-STORAGE SECTION.
77 I             PIC 999 VALUE 0.
77 J             PIC 999 VALUE 0.
77 K             PIC 999 VALUE 0.
77 L             PIC 999 VALUE 0.
77 M             PIC 999 VALUE 0.
77 N             PIC 999 VALUE 0.
01 D01          VALUE "11212223313233343536373839".
   03 D02          OCCURS 13 INDEXED BY D1.
   05 D03          PIC XX.
01 YR1          VALUE "8281807978777675747372".
   03 YR2          OCCURS 11 INDEXED BY Y1.
   05 YR3          PIC XX.
01 SEL          VALUE "1112131415161718192021222324252627".
   03 SL2          OCCURS 17 INDEXED BY S1.
   05 SL3          PIC XX.
01 BUSN         VALUE "11121314151617181920".
   03 BU2          OCCURS 10 INDEXED BY B1.
   05 BU3          PIC XX.
01 DAGU         VALUE "111213141516".
   03 DA2          OCCURS 6 INDEXED BY DA1.
   05 DA3          PIC XX.
01 INCH         VALUE "11121314".
   03 IN2          OCCURS 4 INDEXED BY IN1.

```

	05	IN3	PIC XX.		SVB00560
01	GYGI		VALUE "1112131415161718313233343536373839404142	SVB00570	
-			"4344454647484950".	SVB00580	
	03	GY2	OCCURS 28 INDEXED BY G1.	SVB00590	
	05	GY3	PIC XX.	SVB00600	
01	GANG		VALUE "111213141516313233343536373839404142434445".	SVB00610	
	03	GA2	OCCURS 21 INDEXED BY A1.	SVB00620	
	05	GA3	PIC XX.	SVB00630	
01	CHEK		VALUE "11121331323334353637383940".	SVB00640	
	03	CB2	OCCURS 13 INDEXED BY C1.	SVB00650	
	05	CB3	PIC XX.	SVB00660	
01	CHUM		VALUE "TT010211313233343536373839404142434445".	SVB00670	
	03	CM2	OCCURS 19 INDEXED BY M1.	SVB00680	
	05	CM3	PIC XX.	SVB00690	
01	JUNB		VALUE "111213141531323334353637383940414243".	SVB00700	
	03	JB2	OCCURS 18 INDEXED BY J1.	SVB00710	
	05	JB3	PIC XX.	SVB00720	
01	JUNM		VALUE "TT01020311121314313233343536373839404142	SVB00730	
-			"4344454647484950515253".	SVB00740	
	03	JM2	OCCURS 31 INDEXED BY J2.	SVB00750	
	05	JM3	PIC XX.	SVB00760	
01	KYBK		VALUE "1112131415161731323334353637383940414243	SVB00770	
-			"4445464748495051525354".	SVB00780	
	03	KB2	OCCURS 31 INDEXED BY K1.	SVB00790	
	05	KB3	PIC XX.	SVB00800	
01	KYNM		VALUE "1112131415161718313233343536373839404142	SVB00810	
-			"43444546474849".	SVB00820	
	03	KM2	OCCURS 27 INDEXED BY K2.	SVB00830	
	05	KM3	PIC XX.	SVB00840	
01	JEJU		VALUE "11123132".	SVB00850	
	03	JU2	OCCURS 4 INDEXED BY U1.	SVB00860	
	05	JU3	PIC XX.	SVB00870	
01	TABLE2.			SVB00880	
	03	TAB2	OCCURS 244.	SVB00890	
	05	T2	OCCURS 11 PIC 9(8) COMP-3	SVB00900	
			PROCEDURE DIVISION.	SVB00910	
			OPEN INPUT T-F.	SVB00920	
			OPEN OUTPUT TA-F.	SVB00930	
			MOVE SPACE TO TA-R.	SVB00940	
			PERFORM P1 THRU A2 VARYING I FROM 1 BY 1 UNTIL I > 244.	SVB00950	
			GO TO READ-RTN.	SVB00960	
			P1. PERFORM A3 VARYING J FROM 1 BY 1 UNTIL J > 11.	SVB00970	
			A2. EXIT.	SVB00980	
			A3. MOVE 0 TO T2 (I,J).	SVB00990	
			READ-RTN.	SVB01000	
			READ T-F AT END GO TO PT7-RTN.	SVB01010	
			MOVE 0 TO M.	SVB01020	
			SET D1 TO 1.	SVB01030	
			SEARCH D02 VARYING D1 AT END MOVE 244 TO L GO TO YR7-RTN	SVB01040	
			WHEN D0 = D03 (D1) SET I TO D1 GO TO 10.	SVB01050	
10.				SVB01060	
			GO TO 11D 21D 22D 23D 31D 32D 33D 34D 35D 36D 37D 38D 39D	SVB01070	
			DEPENDING ON I.	SVB01080	
11D.				SVB01090	
			SET S1 TO 1.	SVB01100	

SEARCH SL2 VARYING S1 AT END	SVB01110
MOVE 6 TO J GO TO 11DD	SVB01120
WHEN GUN = SL3 (S1) SET J TO S1 GO TO 11DD.	SVB01130
11DD.	SVB01140
MOVE 2 TO K.	SVB01150
COMPUTE L = K + J.	SVB01160
GO TO YR7-RTN.	SVB01170
21D. SET B1 TO 1.	SVB01180
SEARCH BU2 VARYING B1 AT END	SVB01190
MOVE 3 TO J	SVB01200
WHEN GUN = BU3 (B1) SET J TO B1.	SVB01210
21DD.	SVB01220
MOVE 20 TO K.	SVB01230
COMPUTE L = K + J. GO TO YR7-RTN.	SVB01240
22D. SET DA1 TO 1.	SVB01250
SEARCH DA2 VARYING DA1 AT END	SVB01260
MOVE 3 TO J	SVB01270
WHEN GUN = DA3 (DA1) SET J TO DA1.	SVB01280
22DD.	SVB01290
MOVE 31 TO K.	SVB01300
COMPUTE L = K + J. GO TO YR7-RTN.	SVB01310
23D. SET IN1 TO 1.	SVB01320
SEARCH IN2 VARYING IN1 AT END	SVB01330
MOVE 3 TO J	SVB01340
WHEN GUN = IN3 (IN1) SET J TO IN1.	SVB01350
23DD.	SVB01360
MOVE 38 TO K.	SVB01370
COMPUTE L = K + J. GO TO YR7-RTN.	SVB01380
31D. SET G1 TO 1.	SVB01390
SEARCH GY2 VARYING G1 AT END	SVB01400
MOVE 5 TO J	SVB01410
WHEN GUN = GY3 (G1) SET J TO G1.	SVB01420
31DD.	SVB01430
MOVE 43 TO K.	SVB01440
COMPUTE L = K + J. GO TO YR7-RTN.	SVB01450
32D. SET A1 TO 1.	SVB01460
SEARCH GA2 VARYING A1 AT END	SVB01470
MOVE 1 TO J	SVB01480
WHEN GUN = GA3 (A1) SET J TO A1.	SVB01490
32DD.	SVB01500
MOVE 72 TO K.	SVB01510
COMPUTE L = K + J. GO TO YR7-RTN.	SVB01520
33D. SET C1 TO 1.	SVB01530
SEARCH CB2 VARYING C1 AT END	SVB01540
MOVE 1 TO J	SVB01550
WHEN GUN = CB3 (C1) SET J TO C1.	SVB01560
33DD.	SVB01570
MOVE 94 TO K.	SVB01580
COMPUTE L = K + J. GO TO YR7-RTN.	SVB01590
34D. SET M1 TO 1.	SVB01600
SEARCH CM2 VARYING M1 AT END	SVB01610
MOVE 2 TO J	SVB01620
WHEN GUN = CM3 (M1) SET J TO M1.	SVB01630
34DD.	SVB01640
MOVE 108 TO K.	SVB01650
IF J < 4 MOVE 109 TO M.	
COMPUTE L = K + J. GO TO YR7-RTN.	

35D. SET J1 TO 1.	SVB01660
SEARCH JB2 VARYING J1 AT END	SVB01670
MOVE 1 TO J	SVB01680
WHEN GUN = JB3 (J1) SET J TO J1.	SVB01690
35DD. MOVE 128 TO K.	SVB01700
COMPUTE L = K + J. GO TO YR7-RTN.	SVB01710
36D. SET J2 TO 1.	SVB01720
SEARCH JM2 VARYING J2 AT END	SVB01730
MOVE 1 TO J	SVB01740
WHEN GUN = JM3 (J2) SET J TO J2.	SVB01750
36DD. MOVE 147 TO K.	SVB01760
IF J < 5 MOVE 148 TO M.	SVB01770
COMPUTE L = K + J.	SVB01780
GO TO YR7-RTN.	SVB01790
37D. SET K1 TO 1.	SVB01800
SEARCH KB2 VARYING K1 AT END	SVB01810
MOVE 6 TO J	SVB01820
WHEN GUN = KB3 (K1) SET J TO K1.	SVB01830
37DD. MOVE 179 TO K.	SVB01840
COMPUTE L = K + J.	SVB01850
GO TO YR7-RTN.	SVB01860
38D. SET K2 TO 1.	SVB01870
SEARCH KM2 VARYING K2 AT END	SVB01880
MOVE 1 TO J	SVB01890
WHEN GUN = KM3 (K2) SET J TO K2.	SVB01900
38DD. MOVE 211 TO K.	SVB01910
COMPUTE L = K + J.	SVB01920
GO TO YR7-RTN.	SVB01930
39D. SET U1 TO 1.	SVB01940
SEARCH JU2 VARYING U1 AT END	SVB01950
MOVE 1 TO J	SVB01960
WHEN GUN = JU3 (U1) SET J TO U1.	SVB01970
39DD. MOVE 239 TO K.	SVB01980
COMPUTE L = K + J.	SVB01990
YR7-RTN.	SVB02000
SET Y1 TO 1.	SVB02010
SEARCH YR2 VARYING Y1 AT END GO TO READ-RTN	SVB02020
WHEN YR = YR3 (Y1) SET N TO Y1 GO TO ADD2-RTN.	SVB02030
ADD2-RTN.	SVB02040
IF L = 244 ADD 1 TO T2 (L N) T2 (1 N)	SVB02050
GO TO READ-RTN.	SVB02060
IF M NOT = 0 ADD 1 TO T2(M N).	SVB02070
ADD 1 TO T2 (L N) T2 (K N) T2 (1 N).	SVB02080
GO TO READ-RTN.	SVB02090
ERR-RTN.	SVB02100
DISPLAY " " T-R.	SVB02110
GO TO READ-RTN.	SVB02120
PT7-RTN.	SVB02130
PERFORM P7 VARYING I FROM 1 BY 1 UNTIL I > 244.	SVB02140
CLOSE T-F TA-F.	SVB02150
OPEN INPUT TA-F.	SVB02160
RD-RTN.	SVB02170
READ TA-F AT END CLOSE TA-F STOP RUN.	SVB02180
DISPLAY " " TA-R.	SVB02190
GO TO RD-RTN.	SVB02200

FILE: SVBWT214 TABSMB A1 VM/SP RELEASE 2.1 EXPRESS PUT8302+ SLU203

```
P7. PERFORM P77 VARYING J FROM 1 BY 1 UNTIL J > 11. SVB02210
MOVE BU TO GU. SVB02220
MOVE I TO GU1. SVB02230
MOVE J TO GU2. SVB02240
WRITE TA-R. SVB02250
MOVE SPACE TO TA-R. SVB02260
P77. MOVE T2 (I,J) TO AA (J). SVB02270
/* SVB02280
//GO.TAPEIN DD DSN=SVBWED,UNIT=TAPE,DISP=(OLD,KEEP),VOL=SER=UUPPAK, SVB02290
// LABEL=(2,BLF),DCB=EROPT=SKP SVB02300
//GO.TAPEOUT DD DSN=SVBWMT,UNIT=TAPE,DISP=(NEW,KEEP) SVB02310
//GO.SYSPRINT DD SYSOUT=A SVB02320
//GO.SYSOUT DD SYSOUT=A SVB02330
// SVB02340
```

```

//SVBWD214 JOB TYPRUN=HOLD
/**      BIRTH  DEATH  MARRIAGE  DIVORCE  TAPE  SUM  ADD  UP  ***
//      EXEC  PLIFCLG
//PLI.SYSIN DD *
* PROCESS GS,NEST,OPT(TIME);
  CHOI: PROC OPTIONS(MAIN);
        TAB2=0;
        OPEN FILE(TAPEIN1) INPUT,
              FILE(TAPEIN2) INPUT,
              FILE(TAPEOUT) OUTPUT,
              FILE(SYSPRINT) LINESIZE(132) PAGESIZE(60);
        ON ENDFILE(TAPEIN1) END1_REC='YES';
        ON ENDFILE(TAPEIN2) END2_REC='YES';
        END1_REC = 'NO ';
        END2_REC = 'NO ';
/* TAPE READ PART */
  READ FILE(TAPEIN1) INTO(T_R);
  DO WHILE(END1_REC = 'NO ');
    CALL SET2_R;
    READ FILE(TAPEIN1) INTO(T_R);
  END;
  READ FILE(TAPEIN2) INTO(TI_R);
  DO WHILE(END2_REC = 'NO ');
    CALL SET2_RTN;
    READ FILE(TAPEIN2) INTO(TI_R);
  END;
  CALL WRITE_RTN;
  CLOSE FILE(TAPEIN1),FILE(TAPEIN2),FILE(TAPEOUT);
/* DECLARE PART */
DCL TAPEIN1 FILE RECORD INPUT;
DCL 1 T_R,
    3 GU      PIC '999',
    3 GU1     PIC '99',
    3 GU2     PIC '99',
    3 AA(11)  PIC '(8)9';
DCL TAPEIN2 FILE RECORD INPUT;
DCL 1 TI_R,
    3 KU      PIC '999',
    3 KU1     PIC '99',
    3 KU2     PIC '99',
    3 BB(11)  PIC '(8)9';
DCL TAPEOUT FILE RECORD OUTPUT;
DCL 1 TA_R,
    3 CU      PIC '999',
    3 CU1     PIC '99',
    3 CU2     PIC '99',
    3 CC(11)  PIC '(8)9';
DCL TAB2(244,4,11) FIXED DEC(8);
DCL END1_REC CHAR(3);
DCL END2_REC CHAR(3);
DCL BU      PIC '99';

/**      SUB  PROCEDURE      ***/
  SET2_R: PROC;
    I = GU;

```

SVB00010  
SVB00020  
SVB00030  
SVB00040  
SVB00050  
SVB00060  
SVB00070  
SVB00080  
SVB00090  
SVB00100  
SVB00110  
SVB00120  
SVB00130  
SVB00140  
SVB00150  
SVB00160  
SVB00170  
SVB00180  
SVB00190  
SVB00200  
SVB00210  
SVB00220  
SVB00230  
SVB00240  
SVB00250  
SVB00260  
SVB00270  
SVB00280  
SVB00290  
SVB00300  
SVB00310  
SVB00320  
SVB00330  
SVB00340  
SVB00350  
SVB00360  
SVB00370  
SVB00380  
SVB00390  
SVB00400  
SVB00410  
SVB00420  
SVB00430  
SVB00440  
SVB00450  
SVB00460  
SVB00470  
SVB00480  
SVB00490  
SVB00500  
SVB00510  
SVB00520  
SVB00530  
SVB00540  
SVB00550

```

                IF GU2 = 1 | GU2 = 2 THEN L = GU2;
                                ELSE L = GU2 - 1;
SET1:
    DO J = 1 TO 10,
        K = J + 1;
        TAB2(I,L,K) = AA(J);
    END;
END SET2_R;
SET2_RTN: PROC;
    I=KU;
    IF KU2 = 1 | KU2 = 2 THEN L = KU2;
                                ELSE L = KU2 - 1;
    DO J = 1 TO 11;
        TAB2(I,L,J) = TAB2(I,L,J) + BB(J);
    END;
END SET2_RTN;

WRITE_RTN: PROC;
    CALL SUBWR(TAB2,244,4,11);
END WRITE_RTN;
SUBWR: PROC(ARRAY,II,LL,JJ);
DCL    ARRAY(*,*,*) FIXED DEC(8);
    DO I=1 TO II;
        CU=I;
        DO L = 1 TO LL;
            IF L = 1 | L = 2 THEN CU2 = L;
                                ELSE CU2 = L+1;
            DO J=1 TO JJ;
                CC(J) = ARRAY(I,L,J);
            END;
            WRITE FILE(TAPEOUT) FROM(TA_LR);
        END;
    END SUBWR;
END CHOI;
/*
//GO.TAPEIN1 DD DSN=SVDWED,UNIT=TAPE,DISP=(OLD,KEEP),VOL=SER=UU3035,
//            DCB=(RECFM=FB,LRECL=95,BLKSIZE=1900,EROPT=SKP),
//            LABEL=(2,BLP)
//GO.TAPEIN2 DD DSN=SVDWET,UNIT=TAPE,DISP=(OLD,KEEP),VOL=SER=AUPPAK,
//            DCB=(RECFM=FB,LRECL=95,BLKSIZE=1900,EROPT=SKP),
//            LABEL=(2,BLP)
//GO.TAPEOUT DD DSN=SVDWMT,UNIT=TAPE,DISP=(NEW,KEEP),
//            DCB=(RECFM=FB,LRECL=95,BLKSIZE=1900,EROPT=SKP)
//GO.SYSPRINT DD SYSOUT=A
//GO.SYSOUT   DD SYSOUT=A
//
SVB00560
SVB00570
SVB00580
SVB00590
SVB00600
SVB00610
SVB00620
SVB00630
SVB00640
SVB00650
SVB00660
SVB00670
SVB00680
SVB00690
SVB00700
SVB00710
SVB00720
SVB00730
SVB00740
SVB00750
SVB00760
SVB00770
SVB00780
SVB00790
SVB00800
SVB00810
SVB00820
SVB00830
SVB00840
SVB00850
SVB00860
SVB00870
SVB00880
SVB00890
SVB00900
SVB00910
SVB00920
SVB00930
SVB00940
SVB00950
SVB00960
SVB00970
SVB00980
SVB00990
SVB01000
SVB01010

```



```

//SVBWP214 JOB MSGLEVEL=(1,1),TYPRUN=HOLD
//TABED EXEC COBUCLG,PARM='NOSOURCE'
//COB.SYSIN DD *
ID DIVISION.
PROGRAM-ID.          TA2ED.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SPECIAL-NAMES.
        CO1 IS CHOI
        CSP IS DOUB.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
        SELECT T-F ASSIGN UT-S-TAPEIN.
        SELECT P-F ASSIGN UT-S-PROUT.
DATA DIVISION.
FILE SECTION.
FD T-F          BLOCK CONTAINS 20 RECORDS
                LABEL RECORD STANDARD.

01 T-R.
   03 GU PIC          X(7).
   03 AA OCCURS 11 PIC 9(8).
FD P-F          LABEL RECORD OMITTED.
01 P-R          PIC X(132).
WORKING-STORAGE SECTION.
77 I PIC 9(3) VALUE 0.
77 J PIC 9(3) VALUE 0.
77 K PIC 9(3) VALUE 0.
77 L PIC 9(3) VALUE 0.
77 CNT PIC 99 VALUE 0.
77 ECNT PIC 999 VALUE 0.
77 M PIC 999 VALUE 3.
77 N PIC 999 VALUE 1.
01 LIST2.
   03 F PIC X(22) VALUE SPACE.
   03 BB OCCURS 11 PIC Z(9)9.
01 TAB2 VALUE ZEROS.
   03 TA2 OCCURS 5.
       05 T2 OCCURS 11 PIC 9(8).
PROCEDURE DIVISION.
OPEN INPUT T-F.
OPEN OUTPUT P-F.
** TAB 2 EDITING **
BUSAN.
MOVE SPACE TO P-R.
WRITE P-R AFTER ADVANCING CHOI.
MOVE SPACE TO P-R.
WRITE P-R AFTER ADVANCING 3.
MOVE SPACE TO P-R.
WRITE P-R AFTER ADVANCING 3.
MOVE SPACE TO P-R.
WRITE P-R AFTER ADVANCING 2.
MOVE SPACE TO P-R.
READ-RTN.
READ T-F AT END GO TO END-RTN.
IF ECNT = 800 PERFORM BUSAN MOVE 1 TO CNT

```

SVB00010  
SVB00020  
SVB00030  
SVB00040  
SVB00050  
SVB00060  
SVB00070  
SVB00080  
SVB00090  
SVB00100  
SVB00110  
SVB00120  
SVB00130  
SVB00140  
SVB00150  
SVB00160  
SVB00170  
SVB00180  
SVB00190  
SVB00200  
SVB00210  
SVB00220  
SVB00230  
SVB00240  
SVB00250  
SVB00260  
SVB00270  
SVB00280  
SVB00290  
SVB00300  
SVB00310  
SVB00320  
SVB00330  
SVB00340  
SVB00350  
SVB00360  
SVB00370  
SVB00380  
SVB00390  
SVB00400  
SVB00410  
SVB00420  
SVB00430  
SVB00440  
SVB00450  
SVB00460  
SVB00470  
SVB00480  
SVB00490  
SVB00500  
SVB00510  
SVB00520  
SVB00530  
SVB00540  
SVB00550

```

                ADD 1 TO ECNT                                SVB00560
                MOVE 801 TO N MOVE 803 TO M                SVB00570
                PERFORM PRT7 THRU ET                       SVB00580
                GO TO READ-RTN.                            SVB00590
ADD 1 TO CNT.                                            SVB00600
IF ECNT > 800 IF CNT = 45 PERFORM BUSAN                 SVB00610
                MOVE 1 TO CNT                              SVB00620
                ADD 1 TO ECNT                              SVB00630
                PERFORM PRT7 THRU ET                       SVB00640
                GO TO READ-RTN.                            SVB00650
IF ECNT > 800 ADD 1 TO ECNT                             SVB00660
                PERFORM PRT7 THRU ET                       SVB00670
                GO TO READ-RTN.                            SVB00680
IF CNT = 41 PERFORM BUSAN                               SVB00690
                MOVE 1 TO CNT.                             SVB00700
ADD 1 TO ECNT.                                          SVB00710
PERFORM PRT THRU EX.                                    SVB00720
GO TO READ-RTN.                                         SVB00730
PRT. PERFORM PD                                         VARYING J FROM 1 BY 1 UNTIL J > 11. SVB00740
IF ECNT = N                                             WRITE P-R AFTER ADVANCING 3         SVB00750
                MOVE SPACE TO P-R                          SVB00760
                WRITE P-R FROM LIST2 AFTER ADVANCING 2     SVB00770
                MOVE SPACE TO P-R                          SVB00780
                COMPUTE N = N + 4                           SVB00790
                GO TO EX.                                    SVB00800
IF ECNT = M                                             WRITE P-R FROM LIST2 AFTER ADVANCING 2 SVB00810
                MOVE SPACE TO P-R                          SVB00820
                COMPUTE M = M + 4                           SVB00830
                GO TO EX.                                    SVB00840
WRITE P-R FROM LIST2 AFTER ADVANCING 1.                 SVB00850
MOVE SPACE TO P-R.                                      SVB00860
EX. EXIT.                                               SVB00870
PD. MOVE AA (J) TO BB (J).                              SVB00880
PRT7. PERFORM PD7                                       VARYING J FROM 1 BY 1 UNTIL J > 11. SVB00890
IF ECNT = N                                             WRITE P-R AFTER ADVANCING 2         SVB00900
                MOVE SPACE TO P-R                          SVB00910
                WRITE P-R FROM LIST2 AFTER ADVANCING 2     SVB00920
                MOVE SPACE TO P-R                          SVB00930
                COMPUTE N = N + 4                           SVB00940
                GO TO ET.                                    SVB00950
IF ECNT = M                                             WRITE P-R FROM LIST2 AFTER ADVANCING 2 SVB00960
                MOVE SPACE TO P-R                          SVB00970
                COMPUTE M = M + 4                           SVB00980
                GO TO ET.                                    SVB00990
WRITE P-R FROM LIST2 AFTER ADVANCING 1.                 SVB01000
MOVE SPACE TO P-R.                                      SVB01010
ET. EXIT.                                               SVB01020
PD7. MOVE AA (J) TO BB (J).                              SVB01030
END-RTN.                                                SVB01040
CLOSE T-F P-F.                                         SVB01050
STOP RUN.                                               SVB01060
/*                                                     SVB01070
//GO.TAPEIN DD UNIT=TAPE,DSN=BDMITB,DISP=OLD,VOL=SER=BDMI04, SVB01080
// LABEL=(2,BLP),DCB=(EROPT=SKP)                      SVB01090
//SYSOUT DD SYSOUT=A                                    SVB01100

```

FILE: SVBWP214 PRT2 . A1 VM/SP RELEASE 2.1 EXPRESS PUTB302+ SLU203

//GD.PRCUT DD SYSOUT=A  
//SYSDBCUT DD SYSOUT=A  
//

SVB01110  
SVB01120  
SVB01130

```

//SVBWSM14 JOB TYPRUN=HOLD
/**      BIRTH TABLE SUMMARY      ***
//      EXEC PLIFCLG
//PLI.SYSIN DD *
* PROCESS GS,NEST,OPT(TIME);
  CHOI:  PROC OPTIONS(MAIN);
         TAB3=0;
         TAB4=0;
         TAB5=0;
         TAB6=0;
         OPEN FILE(TAPEIN) INPUT,
         FILE(TAPEOUT) OUTPUT,
         FILE(SYSPRINT) LINESIZE(132) PAGESIZE(60);
         ON ENDFILE(TAPEIN) END_RECORDS = 'YES';
         END_RECORDS = 'NO ';
/*      READ PART */
         READ FILE(TAPEIN) INTO(T_R);
         DO WHILE (END_RECORDS = 'NO ');
           CALL SET_RTN;
         READ FILE(TAPEIN) INTO(T_R);
         END ;
           CALL SUM_RTN ;
           CALL WRITE_RTN;
         CLOSE FILE(TAPEIN),FILE(TAPEOUT);
/*      DECLARE PART */
DCL TAPEIN FILE RECORD INPUT;
DCL 1 T_R,
    3 F1 CHAR(20),
    3 DO PIC '99',
    3 SUN PIC '99',
    3 F2 CHAR(1),
    3 SEX PIC '9',
    3 YR CHAR(2),
    3 WOL PIC '99',
    3 F3 CHAR(14),
    3 EDT PIC '9',
    3 F4 CHAR(4),
    3 BABY PIC '99',
    3 F5 CHAR(12),
    3 AG PIC '99',
    3 F6 CHAR(15);
DCL AGG DEF T_R POS(64) CHAR(2);
DCL TAPEOUT FILE RECORD OUTPUT;
DCL 1 TA_R,
    3 PY0 PIC '99',
    3 PYR PIC '99',
    3 PJUL PIC '999',
    3 PCAN PIC '99',
    3 TAA(40) PIC '(8)9';
DCL YEAR(10) CHAR(2) INIT('82','81','80','79','78','77',
                          '76','75','74','73');
DCL AGE4(49) CHAR(2) INIT(' ',' ',' ',' ',' ',' ',' ',' ',' ',' ',
                          '10','11','12','13','14',' ',' ',
                          '15','16','17','18','19',' ',' ',
                          '20','21','22','23','24',' ',' ');
SVB00010
SVB00020
SVB00030
SVB00040
SVB00050
SVB00060
SVB00070
SVB00080
SVB00090
SVB00100
SVB00110
SVB00120
SVB00130
SVB00140
SVB00150
SVB00160
SVB00170
SVB00180
SVB00190
SVB00200
SVB00210
SVB00220
SVB00230
SVB00240
SVB00250
SVB00260
SVB00270
SVB00280
SVB00290
SVB00300
SVB00310
SVB00320
SVB00330
SVB00340
SVB00350
SVB00360
SVB00370
SVB00380
SVB00390
SVB00400
SVB00410
SVB00420
SVB00430
SVB00440
SVB00450
SVB00460
SVB00470
SVB00480
SVB00490
SVB00500
SVB00510
SVB00520
SVB00530
SVB00540
SVB00550

```

```

                '25','26','27','28','29',' ',' ', SVB00560
                '30','31','32','33','34',' ',' ', SVB00570
                '35','36','37','38','39',' ',' ', SVB00580
                '40','41','42','43','44',' ',' ', SVB00590
                '45','46','47','48','49',' ',' '); SVB00600
DCL AGES(9) PIC '99' INIT(0,15,20,25,30,35,40,45,50); SVB00610
DCL DOD(14) PIC '99' INIT(0,11,21,22,23,31,32,33,34,35,36,37,38,39); SVB00620
DCL TAB3(10,13,3,15) FIXED DEC(8); SVB00630
DCL TAB4(10,5,13,2) FIXED DEC(8); SVB00640
DCL TAB5(10,15,11,13,2) FIXED DEC(8); SVB00650
DCL TAB6(10,5,11,15) FIXED DEC(8); SVB00660
DCL TABM(26) PIC '(8)9'; SVB00670
DCL END_RECORDS CHAR(3); SVB00680
/* SUB PROCEDURE */ SVB00690
SET_RTN: PROCEDURE; SVB00700
    IF SEX= 2 THEN SEX=1; SVB00710
    DO N =1 TO 10; SVB00720
        IF YR = YEAR(N) THEN DO; I=N; SVB00730
            GOTO SET1; SVB00740
        END; SVB00750
    END; SVB00760
    GOTO SET6; SVB00770
SET1: SVB00780
    DO N=1 TO 12; SVB00790
        IF WOL = N THEN DO; J3=N+1; SVB00800
            GOTO SET2; SVB00810
        END; SVB00820
    END; SVB00830
    J3=13; SVB00840
SET2: K3=SEX+1; SVB00850
    DO N=2 TO 14; SVB00860
        IF DO = DOD(N) THEN DO; L3=N; SVB00870
            GOTO SET3; SVB00880
        END; SVB00890
    END; SVB00900
    L3=15; SVB00910
SET3: SVB00920
    IF AG < 15 THEN DO; J4 = 51; GOTO SET4; END; SVB00930
    DO N = 3 TO 49; SVB00940
        IF AGG = AGE4(N) THEN DO; J4=N; SVB00950
            GOTO SET4; SVB00960
        END; SVB00970
    END; SVB00980
    J4=50; SVB00990
SET4: SVB01000
    DO N=1 TO 11; SVB01010
        IF BABY = N THEN DO; K4=N+1; SVB01020
            GOTO SET44; SVB01030
        END; SVB01040
    END; SVB01050
    IF BABY > 11 THEN K4=12; SVB01060
    ELSE K4=13; SVB01070
SET44: SVB01080
    L4=SEX; SVB01090
    J5=L3; SVB01100

```

```

DO N=2 TO 9;
  IF AG < AGE5(N) THEN DO; K5=N;
    GOTO SET5
  END ;
END ;
K5=10;
SET5:
  IF K5 = 2 THEN K5 = 11;
  L5=K4;
  M5=L4;
  DO N=1 TO 4 ;
    IF EDT = N THEN DO; J6=N+1;
      GOTO SET55;
    END ;
  END ;
  J6=3;
SET55:
  K6=K5;
  L6=L3;
  TAB3(I, J3, K3, L3) = TAB3(I, J3, K3, L3)+1;
  TAB3(I, 1, K3, L3) = TAB3(I, 1, K3, L3)+1;
  TAB4(I, J4, K4, L4) = TAB4(I, J4, K4, L4)+1;
  TAB4(I, 1, K4, L4) = TAB4(I, 1, K4, L4)+1;
  TAB5(I, J5, K5, L5, M5) = TAB5(I, J5, K5, L5, M5)+1;
  TAB5(I, 1, K5, L5, M5) = TAB5(I, 1, K5, L5, M5)+1;
  TAB6(I, J6, K6, L6) = TAB6(I, J6, K6, L6)+1;
  TAB6(I, 1, K6, L6) = TAB6(I, 1, K6, L6)+1;
END SET_RTN ;
SET6:
SUM_RTN: PROC;
  TAB3(*, *, 1, *) = TAB3(*, *, 2, *) + TAB3(*, *, 3, *);
  DO I=2 TO 15;
    TAB3(*, *, *, 1) = TAB3(*, *, *, 1) + TAB3(*, *, *, I);
  END;
  DO I=2 TO 44 BY 6;
    DO J=I+1 TO I+5;
      TAB4(*, I, *, *) = TAB4(*, I, *, *) + TAB4(*, J, *, *);
    END;
  END;
  DO I =2 TO 13;
    TAB4(*, *, *, 1, 1) = TAB4(*, *, *, 1, 1) + TAB4(*, *, *, I, 1);
    TAB4(*, *, *, 1, 2) = TAB4(*, *, *, 1, 2) + TAB4(*, *, *, I, 2);
  END;
  DO I=1 TO 15;
    DO J=2 TO 11;
      TAB5(*, I, 1, *, *) = TAB5(*, I, 1, *, *) + TAB5(*, I, J, *, *);
    END;
  END;
  DO I=2 TO 13;
    TAB5(*, *, *, 1, 1) = TAB5(*, *, *, 1, 1) + TAB5(*, *, *, I, 1);
    TAB5(*, *, *, 1, 2) = TAB5(*, *, *, 1, 2) + TAB5(*, *, *, I, 2);
  END;
  DO I=1 TO 5;
    DO J=2 TO 11;
      TAB6(*, I, 1, *) = TAB6(*, I, 1, *) + TAB6(*, I, J, *);
    END;
  END;

```

```

END;
DO I=2 TO 15;
    TAB6(*,*,*,1) = TAB6(*,*,*,1)+TAB6(*,*,*,I);
END;
END SUM_RTN;
WRITE_RTN: PROCEDURE ;
CALL SUBWR (TAB3,3,10,13,3,15);
N=0;
DO I=1 TO 10;
    DO J=1 TO 51;
        PYO=4;
        PYR=I;
        PJUL=J;
        DO K=1 TO 13;
            DO L=1 TO 2;
                PCAN=K;
                N=N+1;
                TAA(N)=TAB4(I,J,K,L);
            END;
        END;
        WRITE FILE(TAPEOUT) FROM (TA_R);
        N=0;
    END;
END;
N=0;
DO I=1 TO 10;
    NN=0;
    DO J=1 TO 15;
        DO K=1 TO 11;
            NN=NN+1;
            PYO=5;
            PYR=I;
            PJUL=NN;
            DO L=1 TO 13;
                DO M=1 TO 2;
                    N=N+1;
                    TAA(N)=TAB5(I,J,K,L,M);
                END;
            END;
        END;
        WRITE FILE(TAPEOUT) FROM (TA_R);
        N=0;
    END;
END;
NN=0;
END;
CALL SUBWR(TAB6,6,10,5,11,15);
END WRITE_RTN;
SUBWR: PROC(ARRAY,PP,II,JJ,KK,LL);
DCL ARRAY(*,*,*,*) FIXED DEC(8);
DO I= 1 TO II;
    NN=0;
    DO J=1 TO JJ;
        DO K=1 TO KK;
            NN=NN+1;
            PYO=PP;

```

SVB01660  
SVB01670  
SVB01680  
SVB01690  
SVB01700  
SVB01710  
SVB01720  
SVB01730  
SVB01740  
SVB01750  
SVB01760  
SVB01770  
SVB01780  
SVB01790  
SVB01800  
SVB01810  
SVB01820  
SVB01830  
SVB01840  
SVB01850  
SVB01860  
SVB01870  
SVB01880  
SVB01890  
SVB01900  
SVB01910  
SVB01920  
SVB01930  
SVB01940  
SVB01950  
SVB01960  
SVB01970  
SVB01980  
SVB01990  
SVB02000  
SVB02010  
SVB02020  
SVB02030  
SVB02040  
SVB02050  
SVB02060  
SVB02070  
SVB02080  
SVB02090  
SVB02100  
SVB02110  
SVB02120  
SVB02130  
SVB02140  
SVB02150  
SVB02160  
SVB02170  
SVB02180  
SVB02190  
SVB02200

```

                PYR=I;
                PJUL=NN;
                DO N=1 TO LL;
                    TAA(N) = ARRAY(I,J,K,N);
                END;
                WRITE FILE(TAPEOUT) FROM(TA_R);
            END;
        END;
    END SUBWR;
    END CHOI;
/*
//GO.TAPEIN DD UNIT=TAPE,DISP=(OLD,KEEP),VOL=SER=CHOI13,
//          DSN=CHOIN1,LABEL=(2,RLP),
//          DCB=(LRECL=80,RECFM=FB,BLKSIZE=3200,EROPT=SKP)
//GO.TAPEOUT DD UNIT=TAPE,DISP=(NEW,KEEP),DSN=CHOIVB,
//           DCB=(LRECL=329,RECFM=FB,BLKSIZE=3290,EROPT=SKP)
//GO.SYSPRINT DD SYSOUT=A
//GO.SYSOUT DD SYSOUT=A
//
SVB02210
SVB02220
SVB02230
SVB02240
SVB02250
SVB02260
SVB02270
SVB02280
SVB02290
SVB02300
SVB02310
SVB02320
SVB02330
SVB02340
SVB02350
SVB02360
SVB02370
SVB02380
SVB02390
SVB02400

```



```

//SVBWDL14 JOB CLASS=V, TYPRUN=HOLD
//*      BIRTH TABLE SUMMARY TO TAPE SUM ADD UP *****
//      EXEC PLIFCLG
//PLI.SYSIN DD *
* PROCESS GS, NEST, OPT(TIME);
CHOI: PROC OPTIONS(MAIN);
      TAB3=0;
      TAB4=0;
      TAB5=0;
      TAB6=0;
      OPEN FILE(TAPEIN1) INPUT,
            FILE(TAPEIN2) INPUT,
            FILE(TAPEOUT) OUTPUT,
            FILE(SYSPRINT) LINESIZE(132) PAGESIZE(60);
      ON ENDFILE(TAPEIN1) END1_REC='YES';
      ON ENDFILE(TAPEIN2) END2_REC='YES';
      END1_REC = 'NO';
      END2_REC = 'NO';
/* TAPE READ PART */
      READ FILE(TAPEIN1) INTO(T_R);
      DO WHILE(END1_REC = 'NO');
        IF GU = 3 THEN CALL SET3_R;
        ELSE
          IF GU = 4 THEN CALL SET4_R;
          ELSE
            IF GU = 5 THEN CALL SET5_R;
            ELSE CALL SET6_R;
            READ FILE(TAPEIN1) INTO(T_R);
          END;
        READ FILE(TAPEIN2) INTO(TI_R);
        DO WHILE(END2_REC = 'NO');
          IF KU = 3 THEN CALL SET3_RTN;
          ELSE
            IF KU = 4 THEN CALL SET4_RTN;
            ELSE
              IF KU = 5 THEN CALL SET5_RTN;
              ELSE CALL SET6_RTN;
              READ FILE(TAPEIN2) INTO(TI_R);
            END;
          CALL WRITE_RTN;
          CLOSE FILE(TAPEIN1), FILE(TAPEIN2), FILE(TAPEOUT);
        /* DECLARE PART */
        DCL TAPEIN1 FILE RECORD INPUT;
        DCL 1 T_R,
            3 GU      PIC '99',
            3 GU1     PIC '99',
            3 GU2     PIC '999',
            3 GU3     PIC '99',
            3 AA(40)  PIC '(8)9';
        DCL TAPEIN2 FILE RECORD INPUT;
        DCL 1 TI_R,
            3 KU      PIC '99',
            3 KU1     PIC '99',
            3 KU2     PIC '999',
            3 KU3     PIC '99',
SVB00010
SVB00020
SVB00030
SVB00040
SVB00050
SVB00060
SVB00070
SVB00080
SVB00090
SVB00100
SVB00110
SVB00120
SVB00130
SVB00140
SVB00150
SVB00160
SVB00170
SVB00180
SVB00190
SVB00200
SVB00210
SVB00220
SVB00230
SVB00240
SVB00250
SVB00260
SVB00270
SVB00280
SVB00290
SVB00300
SVB00310
SVB00320
SVB00330
SVB00340
SVB00350
SVB00360
SVB00370
SVB00380
SVB00390
SVB00400
SVB00410
SVB00420
SVB00430
SVB00440
SVB00450
SVB00460
SVB00470
SVB00480
SVB00490
SVB00500
SVB00510
SVB00520
SVB00530
SVB00540
SVB00550

```

```

        3 BB(40)   PIC '(8)9';
DCL TAPEOUT FILE RECORD OUTPUT;
DCL 1 TA_R,
    3 GUA   PTC '99',
    3 GUB   PIC '99',
    3 GUC   PIC '999',
    3 GUD   PIC '99',
    3 CC(40) PIC '(8)9';
DCL TAB3(10,39,15) FIXED DEC(8);
DCL TAB4(10,51,26) FIXED DEC(8);
DCL TAB5(10,165,26) FIXED DEC(8);
DCL TAB6(10,55,15) FIXED DEC(8);
DCL END1_REC CHAR(3);
DCL END2_REC CHAR(3);
/**** SUB PROCEDURE ****/
SET3_R: PROC;
    IF GU1 = 10 THEN GOTO SET3_E;
    I=GU1+1;
    J=GU2;
    DO K=1 TO 15;
        TAB3(I,J,K) = AA(K);
    END;
SET3_E: END SET3_R;
SET4_R: PROC;
    IF GU1=10 THEN GOTO SET4_E;
    I=GU1+1;
    J=GU2;
    DO K=1 TO 26;
        TAB4(I,J,K) = AA(K);
    END;
SET4_E: END SET4_R;
SET5_R: PROC;
    IF GU1 = 10 THEN GOTO SET5_E;
    I=GU1+1;
    J=GU2;
    DO K=1 TO 26;
        TAB5(I,J,K) = AA(K);
    END;
SET5_E: END SET5_R;
SET6_R: PROC;
    IF GU1 = 10 THEN GOTO SET6_E;
    I= GU1+1;
    J=GU2;
    DO K=1 TO 15;
        TAB6(I,J,K) = AA(K);
    END;
SET6_E: END SET6_R;
SET3_RTN: PROC;
    I=KU1;
    J=KU2;
    DO K=1 TO 15;
        TAB3(I,J,K)=TAB3(I,J,K) + BB(K);
    END;
END SET3_RTN;
SET4_RTN: PROC;

```

SVB00560  
SVB00570  
SVB00580  
SVB00590  
SVB00600  
SVB00610  
SVB00620  
SVB00630  
SVB00640  
SVB00650  
SVB00660  
SVB00670  
SVB00680  
SVB00690  
SVB00700  
SVB00710  
SVB00720  
SVB00730  
SVB00740  
SVB00750  
SVB00760  
SVB00770  
SVB00780  
SVB00790  
SVB00800  
SVB00810  
SVB00820  
SVB00830  
SVB00840  
SVB00850  
SVB00860  
SVB00870  
SVB00880  
SVB00890  
SVB00900  
SVB00910  
SVB00920  
SVB00930  
SVB00940  
SVB00950  
SVB00960  
SVB00970  
SVB00980  
SVB00990  
SVB01000  
SVB01010  
SVB01020  
SVB01030  
SVB01040  
SVB01050  
SVB01060  
SVB01070  
SVB01080  
SVB01090  
SVB01100

```

        I=KU1;
        J=KU2;
        DO K=1 TO 26;
            TAB4(I,J,K)=TAB4(I,J,K) + BB(K);
        END;
    END SET4_RTN;
    SET5_RTN: PROC;
        I=KU1;
        J=KU2;
        DO K=1 TO 26;
            TAB5(I,J,K)= TAB5(I,J,K) + BB(K);
        END;
    END SET5_RTN;
    SET6_RTN: PROC;
        I=KU1;
        J=KU2;
        DO K=1 TO 15;
            TAB6(I,J,K)=TAB6(I,J,K) + BB(K);
        END;
    END SET6_RTN;
    WRITE_RTN: PROC;
        CALL SUBWR(TAB3,10,39,15,3) ;
        CALL SUBWR(TAB4,10,51,26,4) ;
        CALL SUBWR(TAB5,10,165,26,5) ;
        CALL SUBWR(TAB6,10,55,15,6) ;
    END WRITE_RTN;
    SUBWR: PROC (ARRAY,II,JJ,KK,PP);
    DCL ARRAY(*,*,*) FIXED DEC(8);
        DO I=1 TO II;
            DO J=1 TO JJ;
                GUA=PP;
                GUB=I;
                GUC=J;
                DO K=1 TO KK;
                    CC(K) = ARRAY(I,J,K);
                END;
                WRITE FILE(TAPEOUT) FROM(TA_R);
            END;
        END;
    END SUBWR;
END CHOI;
/*
//GO.TAPEIN1 DD DSN=SVDWED,UNIT=TAPE,DISP=(OLD,KEEP),VOL=SER=UU6380,
// DCB=(RECFM=FB,LRECL=329,BLKSIZE=3290,EROPT=SKP),
// LABEL=(2,BLP)
//GO.TAPEIN2 DD DSN=SVDWET,UNIT=TAPE,DISP=(OLD,KEEP),VOL=SER=AU5526,
// DCB=(RECFM=FB,LRECL=329,BLKSIZE=3290,EROPT=SKP),
// LABEL=(2,BLP)
//GO.TAPEOUT DD DSN=SVDWMT,UNIT=TAPE,DISP=(NEW,KEEP),
// DCB=(RECFM=FB,LRECL=329,BLKSIZE=3290,EROPT=SKP)
//GO.SYSPRINT DD SYSOUT=A
//GO.SYSOUT DD SYSOUT=A
//

```

SVB01110  
 SVB01120  
 SVB01130  
 SVB01140  
 SVB01150  
 SVB01160  
 SVB01170  
 SVB01180  
 SVB01190  
 SVB01200  
 SVB01210  
 SVB01220  
 SVB01230  
 SVB01240  
 SVB01250  
 SVB01260  
 SVB01270  
 SVB01280  
 SVB01290  
 SVB01300  
 SVB01310  
 SVB01320  
 SVB01330  
 SVB01340  
 SVB01350  
 SVB01360  
 SVB01370  
 SVB01380  
 SVB01390  
 SVB01400  
 SVB01410  
 SVB01420  
 SVB01430  
 SVB01440  
 SVB01450  
 SVB01460  
 SVB01470  
 SVB01480  
 SVB01490  
 SVB01500  
 SVB01510  
 SVB01520  
 SVB01530  
 SVB01540  
 SVB01550  
 SVB01560  
 SVB01570  
 SVB01580  
 SVB01590  
 SVB01600  
 SVB01610  
 SVB01620  
 SVB01630

```

//SVBWTE14 JOB MSGCLASS=A,CLASS=1,TYPRUN=HOLD
//TABED EXEC COBUCLG,PARM='NOSOURCE'
/** ** TABLE 3 4 5 6 TABLE EDITING **
//COB.SYSIN DD *
ID' DIVISION.
PROGRAM-ID. BIRTED.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SPECIAL-NAMES.
    C01 IS CHOI
    CSP IS DOUB.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT T-F ASSIGN UT-S-TAPEIN.
    SELECT P-F ASSIGN UT-S-PROUT.
DATA DIVISION.
FILE SECTION.
FD T-F BLOCK CONTAINS 10 RECORDS
        LABEL RECORD STANDARD.
01 I-R.
    03 GU PIC 99.
    03 GU1 PIC 99.
    03 CU2 PIC 999.
    03 GU3 PIC 99.
    03 AA OCCURS 40 PIC 9(8).
FD P-F LABEL RECORD OMITTED.
01 P-R PIC X(132).
WORKING-STORAGE SECTION.
77 I PIC 9(3) VALUE 0.
77 J PIC 9(3) VALUE 0.
77 K PIC 9(3) VALUE 0.
77 L PIC 9(3) VALUE 1.
77 M PIC 9(3) VALUE 2.
77 N PIC 9(3) VALUE 3.
77 N1 PIC 9(3) VALUE 0.
77 N2 PIC 9(3) VALUE 0.
77 N3 PIC 9(3) VALUE 0.
77 P PIC 9(3) VALUE 0.
77 B1 PIC 9(3) VALUE 1.
77 B2 PIC 9(3) VALUE 2.
77 B3 PIC 9(3) VALUE 5.
77 B4 PIC 9(3) VALUE 1.
77 S3 PIC 9(3) VALUE 0.
77 S4 PIC 9(3) VALUE 0.
77 S6 PIC 9(3) VALUE 0.
77 CNT PIC 9(3) VALUE 0.
77 D1 PIC 99 VALUE 0.
01 LIST3.
    03 F PIC X(13) VALUE SPACE.
    03 WHL3 PIC Z(7)9.
    03 P3 OCCURS 12 PIC Z(7)9.
    03 P31 PIC Z(6)9.
    03 P32 PIC Z(5)9.
01 LIST41.
    03 F PIC X(32) VALUE SPACE.

```

SVB00010  
SVB00020  
SVB00030  
SVB00040  
SVB00050  
SVB00060  
SVB00070  
SVB00080  
SVB00090  
SVB00100  
SVB00110  
SVB00120  
SVB00130  
SVB00140  
SVB00150  
SVB00160  
SVB00170  
SVB00180  
SVB00190  
SVB00200  
SVB00210  
SVB00220  
SVB00230  
SVB00240  
SVB00250  
SVB00260  
SVB00270  
SVB00280  
SVB00290  
SVB00300  
SVB00310  
SVB00320  
SVB00330  
SVB00340  
SVB00350  
SVB00360  
SVB00370  
SVB00380  
SVB00390  
SVB00400  
SVB00410  
SVB00420  
SVB00430  
SVB00440  
SVB00450  
SVB00460  
SVB00470  
SVB00480  
SVB00490  
SVB00500  
SVB00510  
SVB00520  
SVB00530  
SVB00540  
SVB00550

	03	WHL4	PIC Z(11)9.	SVB00560
	03	WHLM	PIC Z(11)9.	SVB00570
	03	WHLF	PIC Z(11)9.	SVB00580
	03	P41	OCCURS 8 PIC Z(7)9.	SVB00590
01		LIST42.		SVB00600
	03	P42	OCCURS 16 PIC Z(7)9.	SVB00610
01		LIST51.		SVB00620
	03	F	PIC X(32) VALUE SPACE.	SVB00630
	03	WHL5	PIC Z(11)9.	SVB00640
	03	WH5M	PIC Z(9)9.	SVB00650
	03	WH5F	PIC Z(9)9.	SVB00660
	03	P51	OCCURS 8 PIC Z(7)9.	SVB00670
01		LIST52.		SVB00680
	03	P52	OCCURS 16 PIC Z(7)9.	SVB00690
01		LIST6.		SVB00700
	03	F	PIC X(15) VALUE SPACE.	SVB00710
	03	WHL6	PIC Z(7)9.	SVB00720
	03	P7	OCCURS 12 PIC Z(7)9.	SVB00730
	03	P71	PIC Z(6)9.	SVB00740
	03	P72	PIC Z(5)9.	SVB00750
01		TABL4.		SVB00760
	03	TAB4	OCCURS 51.	SVB00770
	05	T4	OCCURS 27 PIC 9(8).	SVB00780
01		TABL5.		SVB00790
	03	TAB5	OCCURS 165.	SVB00800
	05	T5	OCCURS 27 PIC 9(8).	SVB00810
01		LDF	PIC X(30) VALUE '050811161922273033384144495255'.	SVB00820
01		LLDF	REDEFINES LDF.	SVB00830
	03	LSKIP	OCCURS 15 PIC 99.	SVB00840
		PROCEDURE	DIVISION.	SVB00850
		OPEN	INPUT T-F.	SVB00860
		OPEN	OUTPUT P-F.	SVB00870
		BUSAN.		SVB00880
		MOVE	SPACE TO P-R.	SVB00890
		WRITE	P-R AFTER ADVANCING CHOI.	SVB00900
		MOVE	SPACE TO P-R.	SVB00910
		WRITE	P-R AFTER ADVANCING 3.	SVB00920
		MOVE	SPACE TO P-R.	SVB00930
		WRITE	P-R AFTER ADVANCING 3.	SVB00940
		MOVE	SPACE TO P-R.	SVB00950
		WRITE	P-R AFTER ADVANCING 3.	SVB00960
		MOVE	SPACE TO P-R.	SVB00970
		WRITE	P-R AFTER ADVANCING 3.	SVB00980
		MOVE	SPACE TO P-R.	SVB00990
		WRITE	P-R AFTER ADVANCING 2.	SVB01000
		MOVE	SPACE TO P-R.	SVB01010
		READ-RTN.		SVB01020
		READ	T-F AT END GO TO CLOSE-RTN.	SVB01030
		COMPUTE	D1 = GU - 2.	SVB01040
		GO TO	3T 4T 5T 6T DEPENDING ON D1.	SVB01050
3T.				SVB01060
		IF	S3 = 1 GO TO 3T-PRT.	SVB01070
		MOVE	GU1 TO B1.	SVB01080
		MOVE	1 TO S3.	SVB01090
3T-PRT.				SVB01100

```

IF GU1 NOT = B1      PERFORM BUSAN                      SVB01110
                    MOVE GU1 TO B1                    SVB01120
                    MOVE 1 TO L.                      SVB01130
MOVE AA (1) TO WHL3.                               SVB01140
PERFORM PR3      VARYING I FROM 2 BY 1 UNTIL I > 13.  SVB01150
MOVE AA(14) TO P31.                               SVB01160
MOVE AA(15) TO P32.                               SVB01170
IF GU2 = L      WRITE P-R FROM LIST3 AFTER ADVANCING 3  SVB01180
                WRITE P-R FROM LIST3 AFTER ADVANCING DOUB SVB01190
                MOVE SPACE TO P-R                    SVB01200
                COMPUTE L = L + 3                     SVB01210
                GO TO READ-RTN.                       SVB01220
WRITE P-R FROM LIST3 AFTER ADVANCING 2.           SVB01230
MOVE SPACE TO P-R.                               SVB01240
GO TO READ-RTN.                                  SVB01250
PR3. COMPUTE J = I - 1.                           SVB01260
     MOVE AA (I) TO P3 (J).                       SVB01270
4T. IF GU4 NOT = B4      GO TO 4PR-RTN.            SVB01280
RT. MOVE GU1 TO P.                                  SVB01290
     MOVE GU2 TO I.                                 SVB01300
     COMPUTE T4 (I 1) = AA (1) + AA (2).           SVB01310
     PERFORM PR4      VARYING J FROM 1 BY 1 UNTIL J > 26. SVB01320
     GO TO READ-RTN.                               SVB01330
PR4.                                                     SVB01340
     COMPUTE K = J + 1.                             SVB01350
     MOVE AA (J) TO T4 (I K).                       SVB01360
4PR-RTN.                                             SVB01370
     MOVE 2 TO M. MOVE 3 TO N.                       SVB01380
     PERFORM BUSAN.                                  SVB01390
     PERFORM PTA THRU ET      VARYING I FROM 1 BY 1 UNTIL I > 51. SVB01400
     MOVE 2 TO M.                                     SVB01410
     MOVE 3 TO N.                                     SVB01420
     PERFORM BUSAN.                                  SVB01430
     PERFORM PTB THRU EX      VARYING I FROM 1 BY 1 UNTIL I > 51. SVB01440
BB. MOVE GU1 TO B4.                                  SVB01450
     GO TO RT.                                       SVB01460
PTA.                                                     SVB01470
     MOVE T4 (I 1) TO WHL4.                           SVB01480
     MOVE T4 (I 2) TO WHLM.                           SVB01490
     MOVE T4 (I 3) TO WHLF.                           SVB01500
     PERFORM PA      VARYING J FROM 4 BY 1 UNTIL J > 11.  SVB01510
     IF I = 1      WRITE P-R FROM LIST41 AFTER ADVANCING 3  SVB01520
                   WRITE P-R FROM LIST41 AFTER ADVANCING DOUB SVB01530
                   MOVE SPACE TO P-R                SVB01540
                   GO TO ET.                          SVB01550
     IF I = M      WRITE P-R FROM LIST41 AFTER ADVANCING 3  SVB01560
                   WRITE P-R FROM LIST41 AFTER ADVANCING DOUB SVB01570
                   COMPUTE M = M + 6                 SVB01580
                   MOVE SPACE TO P-R                SVB01590
                   GO TO ET.                          SVB01600
     IF I = N      WRITE P-R FROM LIST41 AFTER ADVANCING 3  SVB01610
                   MOVE SPACE TO P-R                SVB01620
                   COMPUTE N = N + 6                 SVB01630
                   GO TO ET.                          SVB01640
WRITE P-R FROM LIST41 AFTER ADVANCING 1.           SVB01650

```

MOVE SPACE TO P-R.	SVB01660
ET. EXIT.	SVB01670
PA. COMPUTE K = J - 3.	SVB01680
MOVE T4 (I J) TO P41 (K).	SVB01690
PTB.	SVB01700
PERFORM PB VARYING J FROM 12 BY 1 UNTIL J > 27.	SVB01710
IF I = 1 WRITE P-R FROM LIST42 AFTER ADVANCING 3	SVB01720
WRITE P-R FROM LIST42 AFTER ADVANCING DOUB	SVB01730
MOVE SPACE TO P-R	SVB01740
GO TO EX.	SVB01750
IF I = M WRITE P-R FROM LIST42 AFTER ADVANCING 3	SVB01760
WRITE P-R FROM LIST42 AFTER ADVANCING DOUB	SVB01770
MOVE SPACE TO P-R	SVB01780
COMPUTE M = M + 6	SVB01790
GO TO EX.	SVB01800
IF I = N WRITE P-R FROM LIST42 AFTER ADVANCING 3	SVB01810
MOVE SPACE TO P-R	SVB01820
COMPUTE N = N + 6	SVB01830
GO TO EX.	SVB01840
WRITE P-R FROM LIST42 AFTER ADVANCING 1.	SVB01850
MOVE SPACE TO P-R.	SVB01860
EX. EXIT.	SVB01870
PB. COMPUTE K = J - 11.	SVB01880
MOVE T4 (I J) TO P42 (K).	SVB01890
5T.	SVB01900
IF S4 = 1 GO TO 5A.	SVB01910
PERFORM 4PR-RTN.	SVB01920
MOVE GU1 TO B1.	SVB01930
MOVE 1 TO S4.	SVB01940
5A.	SVB01950
IF GU1 NOT = B1 GO TO 5PRT-RTN.	SVB01960
5T-RETURN.	SVB01970
MOVE GU1 TO P.	SVB01980
MOVE GU2 TO I.	SVB01990
COMPUTE T5 (I 1) = AA (1) + AA (2).	SVB02000
PERFORM PR5 VARYING J FROM 1 BY 1 UNTIL J > 26.	SVB02010
GO TO READ-RTN.	SVB02020
PR5.	SVB02030
COMPUTE K = J + 1.	SVB02040
MOVE AA (J) TO T5 (I K).	SVB02050
5PRT-RTN.	SVB02060
MOVE 1 TO L. MOVE 2 TO M.	SVB02070
MOVE 5 TO N1.	SVB02080
MOVE 8 TO N2.	SVB02090
MOVE 11 TO N3.	SVB02100
PERFORM BUSAN.	SVB02110
PERFORM PT5 THRU EJ VARYING I FROM 1 BY 1 UNTIL I > 55.	SVB02120
MOVE 1 TO L.	SVB02130
MOVE 2 TO M.	SVB02140
MOVE 5 TO N1.	SVB02150
MOVE 8 TO N2.	SVB02160
MOVE 11 TO N3.	SVB02170
PERFORM BUSAN.	SVB02180
PERFORM PTC THRU EK VARYING I FROM 1 BY 1 UNTIL I > 55.	SVB02190
MOVE 56 TO L.	SVB02200

MOVE 57 TO M.		SVB02210
MOVE 60 TO N1.		SVB02220
MOVE 63 TO N2.		SVB02230
MOVE 66 TO N3.		SVB02240
PERFORM BUSAN.		SVB02250
PERFORM PTD THRU X1	VARYING I FROM 56 BY 1 UNTIL I > 110.	SVB02260
MOVE 56 TO L.		SVB02270
MOVE 57 TO M.		SVB02280
MOVE 60 TO N1.		SVB02290
MOVE 63 TO N2.		SVB02300
MOVE 66 TO N3.		SVB02310
PERFORM BUSAN.		SVB02320
PERFORM PTE THRU X2	VARYING I FROM 56 BY 1 UNTIL I > 110.	SVB02330
MOVE 111 TO L.		SVB02340
MOVE 112 TO M.		SVB02350
MOVE 115 TO N1.		SVB02360
MOVE 118 TO N2.		SVB02370
MOVE 121 TO N3.		SVB02380
PERFORM BUSAN.		SVB02390
PERFORM PTF THRU X3	VARYING I FROM 111 BY 1 UNTIL I > 165.	SVB02400
MOVE 111 TO L.		SVB02410
MOVE 112 TO M.		SVB02420
MOVE 115 TO N1.		SVB02430
MOVE 118 TO N2.		SVB02440
MOVE 121 TO N3.		SVB02450
PERFORM BUSAN.		SVB02460
PERFORM PTG THRU X4	VARYING I FROM 111 BY 1 UNTIL I > 165.	SVB02470
		SVB02480
CC. MOVE GU1 TO B1.		SVB02490
GO TO 5T-RETURN.		SVB02500
PT5.		SVB02510
MOVE T5 (I 1) TO WHL5.		SVB02520
MOVE T5 (I 2) TO WH5M.		SVB02530
MOVE T5 (I 3) TO WH5F.		SVB02540
PERFORM P5	VARYING J FROM 4 BY 1 UNTIL J > 11.	SVB02550
IF I = L	WRITE P-R FROM LIST51 AFTER ADVANCING 3	SVB02560
	WRITE P-R FROM LIST51 AFTER ADVANCING DOUB	SVB02570
MOVE SPACE TO P-R		SVB02580
COMPUTE L = L + 11		SVB02590
GO TO EJ.		SVB02600
IF I = M	WRITE P-R FROM LIST51 AFTER ADVANCING 3	SVB02610
	MOVE SPACE TO P-R	SVB02620
	COMPUTE M = M + 11	SVB02630
	GO TO EJ.	SVB02640
IF I = N1	WRITE P-R FROM LIST51 AFTER ADVANCING 2	SVB02650
	MOVE SPACE TO P-R	SVB02660
	COMPUTE N1 = N1 + 11	SVB02670
	GO TO EJ.	SVB02680
IF I = N2	WRITE P-R FROM LIST51 AFTER ADVANCING 2	SVB02690
	MOVE SPACE TO P-R	SVB02700
	COMPUTE N2 = N2 + 11	SVB02710
	GO TO EJ.	SVB02720
IF I = N3	WRITE P-R FROM LIST51 AFTER ADVANCING 2	SVB02730
	MOVE SPACE TO P-R	SVB02740
	COMPUTE N3 = N3 + 11	SVB02750



	GO TO EJ.	SVB02760
	WRITE P-R FROM LIST51 AFTER ADVANCING 1.	SVB02770
	MOVE SPACE TO P-R.	SVB02780
EJ.	EXIT.	SVB02790
P5.	COMPUTE K = J - 3.	SVB02800
	MOVE T5 (I J) TO P51 (K).	SVB02810
PTC.		SVB02820
	PERFORM PC VARYING J FROM 12 BY 1 UNTIL J > 27.	SVB02830
	IF I = L WRITE P-R FROM LIST52 AFTER ADVANCING 3	SVB02840
	WRITE P-R FROM LIST52 AFTER ADVANCING DOUB	SVB02850
	MOVE SPACE TO P-R	SVB02860
	COMPUTE L = L + 11	SVB02870
	GO TO EK.	SVB02880
	IF I = M WRITE P-R FROM LIST52 AFTER ADVANCING 3	SVB02890
	MOVE SPACE TO P-R	SVB02900
	COMPUTE M = M + 11	SVB02910
	GO TO EK.	SVB02920
	IF I = N1 WRITE P-R FROM LIST52 AFTER ADVANCING 2	SVB02930
	MOVE SPACE TO P-R	SVB02940
	COMPUTE N1 = N1 + 11	SVB02950
	GO TO EK.	SVB02960
	IF I = N2 WRITE P-R FROM LIST52 AFTER ADVANCING 2	SVB02970
	MOVE SPACE TO P-R	SVB02980
	COMPUTE N2 = N2 + 11	SVB02990
	GO TO EK.	SVB03000
	IF I = N3 WRITE P-R FROM LIST52 AFTER ADVANCING 2	SVB03010
	MOVE SPACE TO P-R	SVB03020
	COMPUTE N3 = N3 + 11	SVB03030
	GO TO EK.	SVB03040
	WRITE P-R FROM LIST52 AFTER ADVANCING 1.	SVB03050
	MOVE SPACE TO P-R.	SVB03060
EK.	EXIT.	SVB03070
PC.	COMPUTE K = J - 11.	SVB03080
	MOVE T5 (I J) TO P52 (K).	SVB03090
PTD.	MOVE T5 (I 1) TO WHL5.	SVB03100
	MOVE T5 (I 2) TO WH5M.	SVB03110
	MOVE T5 (I 3) TO WH5F.	SVB03120
	PERFORM PD VARYING J FROM 4 BY 1 UNTIL J > 11.	SVB03130
	IF I = L WRITE P-R FROM LIST51 AFTER ADVANCING 3	SVB03140
	WRITE P-R FROM LIST51 AFTER ADVANCING DOUB	SVB03150
	MOVE SPACE TO P-R	SVB03160
	COMPUTE L = L + 11	SVB03170
	GO TO X1.	SVB03180
	IF I = M WRITE P-R FROM LIST51 AFTER ADVANCING 3	SVB03190
	MOVE SPACE TO P-R	SVB03200
	COMPUTE M = M + 11	SVB03210
	GO TO X1.	SVB03220
	IF I = N1 WRITE P-R FROM LIST51 AFTER ADVANCING 2	SVB03230
	MOVE SPACE TO P-R	SVB03240
	COMPUTE N1 = N1 + 11	SVB03250
	GO TO X1.	SVB03260
	IF I = N2 WRITE P-R FROM LIST51 AFTER ADVANCING 2	SVB03270
	MOVE SPACE TO P-R	SVB03280
	COMPUTE N2 = N2 + 11	SVB03290
	GO TO X1.	SVB03300

IF I = N3	WRITE P-R FROM LIST51	AFTER ADVANCING 2	SVB03310
	MOVE SPACE TO P-R		SVB03320
	COMPUTE N3 = N3 + 11		SVB03330
	GO TO X1.		SVB03340
	WRITE P-R FROM LIST51	AFTER ADVANCING 1.	SVB03350
	MOVE SPACE TO P-R.		SVB03360
X1.	EXIT.		SVB03370
PD.	COMPUTE K = J - 3.		SVB03380
	MOVE T5 (I J) TO P51 (K).		SVB03390
PTE.			SVB03400
PERFORM PE	VARYING J FROM 12 BY 1 UNTIL J > 27.		SVB03410
IF I = L	WRITE P-R FROM LIST52	AFTER ADVANCING 3	SVB03420
	WRITE P-R FROM LIST52	AFTER ADVANCING DOUB	SVB03430
	MOVE SPACE TO P-R		SVB03440
	COMPUTE L = L + 11		SVB03450
	GO TO X2.		SVB03460
IF I = M	WRITE P-R FROM LIST52	AFTER ADVANCING 3	SVB03470
	MOVE SPACE TO P-R		SVB03480
	COMPUTE M = M + 11		SVB03490
	GO TO X2.		SVB03500
IF I = N1	WRITE P-R FROM LIST52	AFTER ADVANCING 2	SVB03510
	MOVE SPACE TO P-R		SVB03520
	COMPUTE N1 = N1 + 11		SVB03530
	GO TO X2.		SVB03540
IF I = N2	WRITE P-R FROM LIST52	AFTER ADVANCING 2	SVB03550
	MOVE SPACE TO P-R		SVB03560
	COMPUTE N2 = N2 + 11		SVB03570
	GO TO X2.		SVB03580
IF I = N3	WRITE P-R FROM LIST52	AFTER ADVANCING 2	SVB03590
	MOVE SPACE TO P-R		SVB03600
	COMPUTE N3 = N3 + 11		SVB03610
	GO TO X2.		SVB03620
	WRITE P-R FROM LIST52	AFTER ADVANCING 1.	SVB03630
	MOVE SPACE TO P-R.		SVB03640
X2.	EXIT.		SVB03650
PE.	COMPUTE K = J - 11.		SVB03660
	MOVE T5 (I J) TO P52 (K).		SVB03670
PTF.	MOVE T5 (I 1) TO WHL5.		SVB03680
	MOVE T5 (I 2) TO WH5M.		SVB03690
	MOVE T5 (I 3) TO WH5F.		SVB03700
PERFORM PFF	VARYING J FROM 4 BY 1 UNTIL J > 11.		SVB03710
IF I = L	WRITE P-R FROM LIST51	AFTER ADVANCING 3	SVB03720
	WRITE P-R FROM LIST51	AFTER ADVANCING DOUB	SVB03730
	MOVE SPACE TO P-R		SVB03740
	COMPUTE L = L + 11		SVB03750
	GO TO X3.		SVB03760
IF I = M	WRITE P-R FROM LIST51	AFTER ADVANCING 3	SVB03770
	MOVE SPACE TO P-R		SVB03780
	COMPUTE M = M + 11		SVB03790
	GO TO X3.		SVB03800
IF I = N1	WRITE P-R FROM LIST51	AFTER ADVANCING 2	SVB03810
	MOVE SPACE TO P-R		SVB03820
	COMPUTE N1 = N1 + 11		SVB03830
	GO TO X3.		SVB03840
IF I = N2	WRITE P-R FROM LIST51	AFTER ADVANCING 2	SVB03850

	MOVE SPACE TO P-R	SVB03860
	COMPUTE N2 = N2 + 11	SVB03870
	GO TO X3.	SVB03880
IF I = N3	WRITE P-R FROM LIST51 AFTER ADVANCING 2	SVB03890
	MOVE SPACE TO P-R	SVB03900
	COMPUTE N3 = N3 + 11	SVB03910
	GO TO X3.	SVB03920
	WRITE P-R FROM LIST51 AFTER ADVANCING 1.	SVB03930
	MOVE SPACE TO P-R.	SVB03940
X3. EXIT.		SVB03950
PFF.		SVB03960
	COMPUTE K = J - 3.	SVB03970
	MOVE T5 (I J) TO P51 (K).	SVB03980
PTG.		SVB03990
	PERFORM PGG	SVB04000
	VARYING J FROM 12 BY 1 UNTIL J > 27.	SVB04000
IF I = L	WRITE P-R FROM LIST52 AFTER ADVANCING 3	SVB04010
	WRITE P-R FROM LIST52 AFTER ADVANCING DOUB	SVB04020
	MOVE SPACE TO P-R	SVB04030
	COMPUTE L = L + 11	SVB04040
	GO TO X4.	SVB04050
IF I = M	WRITE P-R FROM LIST52 AFTER ADVANCING 3	SVB04060
	MOVE SPACE TO P-R	SVB04070
	COMPUTE M = M + 11	SVB04080
	GO TO X4.	SVB04090
IF I = N1	WRITE P-R FROM LIST52 AFTER ADVANCING 2	SVB04100
	MOVE SPACE TO P-R	SVB04110
	COMPUTE N1 = N1 + 11	SVB04120
	GO TO X4.	SVB04130
IF I = N2	WRITE P-R FROM LIST52 AFTER ADVANCING 2	SVB04140
	MOVE SPACE TO P-R	SVB04150
	COMPUTE N2 = N2 + 11	SVB04160
	GO TO X4.	SVB04170
IF I = N3	WRITE P-R FROM LIST52 AFTER ADVANCING 2	SVB04180
	MOVE SPACE TO P-R	SVB04190
	COMPUTE N3 = N3 + 11	SVB04200
	GO TO X4.	SVB04210
	WRITE P-R FROM LIST52 AFTER ADVANCING 1.	SVB04220
	MOVE SPACE TO P-R.	SVB04230
X4. EXIT.		SVB04240
PGG.		SVB04250
	COMPUTE K = J - 11.	SVB04260
	MOVE T5 (I J) TO P52 (K).	SVB04270
6T.		SVB04280
	IF S6 = 1 GO TO 6PRT-RTN.	SVB04290
	PERFORM 5PRT-RTN.	SVB04300
	MOVE 1 TO L.	SVB04310
	MOVE 2 TO M.	SVB04320
	MOVE GU1 TO B1.	SVB04330
	MOVE 1 TO S6.	SVB04340
	PERFORM BUSAN.	SVB04350
6PRT-RTN.		SVB04360
	IF GU1 NOT = B1 PERFORM BUSAN	SVB04370
	MOVE GU1 TO B1	SVB04380
	MOVE 1 TO L	SVB04390
	MOVE 2 TO M.	SVB04400

```

MOVE AA (1) TO WHL6. SVB04410
PERFORM PR6 VARYING I FROM 2 BY 1 UNTIL I > 13. SVB04420
MOVE AA(14) TO P71. SVB04430
MOVE AA(15) TO P72. SVB04440
IF GU2 = L WRITE P-R FROM LIST6 AFTER ADVANCING 3 SVB04450
WRITE P-R FROM LIST6 AFTER ADVANCING DOUB SVB04460
MOVE SPACE TO P-R SVB04470
COMPUTE L = L + 11 SVB04480
GO TO READ-RTN. SVB04490
IF GU2 = M WRITE P-R FROM LIST6 AFTER ADVANCING 3 SVB04500
MOVE SPACE TO P-R SVB04510
COMPUTE M = M + 11 SVB04520
GO TO READ-RTN. SVB04530
PERFORM A6 VARYING J FROM 1 BY 1 UNTIL J > 15. SVB04540
WRITE P-R FROM LIST6 AFTER ADVANCING 1. SVB04550
MOVE SPACE TO P-R. SVB04560
GO TO READ-RTN. SVB04570
A6. IF GU2 = LSKIP (J) GO TO TL6. SVB04580
TL6. WRITE P-R FROM LIST6 AFTER ADVANCING 2. SVB04590
MOVE SPACE TO P-R. SVB04600
GO TO READ-RTN. SVB04610
PR6. SVB04620
COMPUTE K = I - 1. SVB04630
MOVE AA (I) TO P7(K). SVB04640
CLOSE-RTN. SVB04650
CLOSE T-F P-F. SVB04660
STOP RUN. SVB04670
SVB04680
/* SVB04690
//GO.TAPEIN DD UNIT=TAPE,DISP=(OLD,KEEP),DSN=VBSM13,VOL=SER=VB1111, SVB04700
// LABEL=(2,BLP),DCB=(EROPT=SKP) SVB04710
//SYSOUT DD SYSOUT=A SVB04720
//GO.PROUT DD SYSOUT=A SVB04730
//SYSDBOUT DD SYSOUT=A SVB04740
//

```

//SVDWTT14 JOB TYPRUN=HOLD	SVD00010
//TTT EXEC COBUCLG	SVD00020
//COB.SYSIN DD *	SVD00030
ID DIVISION.	SVD00040
PROGRAM-ID. TTT.	SVD00050
ENVIRONMENT DIVISION.	SVD00060
INPUT-OUTPUT SECTION.	SVD00070
FILE-CONTROL.	SVD00080
SELECT T-F ASSIGN UT-S-TAPEIN.	SVD00090
SELECT TA-F ASSIGN UT-S-TAPEOUT.	SVD00100
DATA DIVISION.	SVD00110
FILE SECTION.	SVD00120
FD T-F LABEL RECORD OMITTED.	SVD00130
01 T-R.	SVD00140
03 AA PIC X(13).	SVD00150
03 BB PIC X(38).	SVD00160
03 CC PIC X(3).	SVD00170
03 DD REDEFINES CC PIC 999.	SVD00180
03 F PIC X(26).	SVD00190
FD TA-F BLOCK CONTAINS 40 RECORDS	SVD00200
LABEL RECORD STANDARD.	SVD00210
01 TA-R.	SVD00220
02 TAA.	SVD00230
03 TA PIC X(13).	SVD00240
03 TC PIC 999.	SVD00250
02 TBB.	SVD00260
03 TB PIC X(38).	SVD00270
03 F PIC X(26).	SVD00280
WORKING-STORAGE SECTION.	SVD00290
77 ECNT PIC 9(7) VALUE 0.	SVD00300
77 OCNT PIC 9(7) VALUE 0.	SVD00310
PROCEDURE DIVISION.	SVD00320
OPEN INPUT T-F.	SVD00330
OPEN OUTPUT TA-F.	SVD00340
MOVE SPACE TO TA-R.	SVD00350
READ-RTN.	SVD00360
READ T-F AT END GO TO END-RTN.	SVD00370
ADD 1 TO ECNT.	SVD00380
EXAMINE CC TALLYING ALL SPACE REPLACING BY 0.	SVD00390
IF DD NOT NUMERIC GO TO MOVE-RTN.	SVD00400
COMPUTE TC = DD / 10 ** TALLY.	SVD00410
MOVE-RTN.	SVD00420
MOVE AA TO TA.	SVD00430
MOVE BB TO TB.	SVD00440
WRITE TA-R.	SVD00450
MOVE SPACE TO TA-R.	SVD00460
ADD 1 TO OCNT.	SVD00470
GO TO READ-RTN.	SVD00480
END-RTN.	SVD00490
DISPLAY " ECNT = " ECNT.	SVD00500
DISPLAY " OCNT = " OCNT.	SVD00510
CLOSE T-F TA-F.	SVD00520
OPEN INPUT TA-F.	SVD00530
RD-RTN.	SVD00540
READ TA-F AT END CLOSE TA-F STOP RUN.	SVD00550

FILE: SVDWTT14 RJTTT A1 VM/SP RELEASE 2.1 EXPRESS PUT8302+ SLU203

DISPLAY " " TAA.	SVD00560
EX. PERFORM RD-RTN 2000 TIMES.	SVD00570
CLOSE TA-F.	SVD00580
STOP RUN.	SVD00590
/*	SVD00600
//GO.TAPEIN DD DSN=VBCHOI,UNIT=TAPE,DISP=OLD,VOL=SER=000001,	SVD00610
// DCB=EROPT=SKP,LABEL=(,NL)	SVD00620
// DD DSN=HB,UNIT=AF=TAPEIN,LABEL=(,NL),VOL=SER=000002	SVD00630
// DD DSN=HB,UNIT=AF=TAPEIN,LABEL=(,NL),VOL=SER=000003	SVD00640
//GO.TAPEOUT DD DSN=CHOI13,UNIT=TAPE,DISP=NEW,VOL=SER=SCRATCH	SVD00650
//SYSOUT DD SYSOUT=A	SVD00660
//	SVD00670

```

//SVDMSM14 JOB TYPRUN=HOLD
/** DEATH TABLE SUMMARY ***
// EXEC PLIFCLG
//PLI.SYSIN DD *
* PROCESS GS,NEST,OPT(TIME);
  CHOI: PROC OPTIONS(MAIN);
        TAB7=0;
        TAB8=0;
        TAB9=0;
        OPEN FILE(TAPEIN) INPUT,
        FILE(TAPEOUT) OUTPUT,
        FILE(SYSPRINT) LINESIZE(132) PAGESIZE(60);
        ON ENDFILE(TAPEIN) END_RECORDS = 'YES';
        END_RECORDS = 'NO ';
/*
  READ PART /*
  READ FILE(TAPEIN) INTO(T_R);
  DO WHILE (END_RECORDS = 'NO ');
    CALL SET_RTN;
  READ FILE(TAPEIN) INTO(T_R);
  END ;
    CALL SUM_RTN ;
    CALL WRITE_RTN;
  CLOSE FILE(TAPEIN),FILE(TAPEOUT);
/*
  DECLARE PART /*
DCL TAPEIN FILE RECORD INPUT;
DCL 1 T_R,
    3 F1 CHAR(20),
    3 DO PIC '99',
    3 GUN PIC '99',
    3 SEX PIC '9',
    3 F2 CHAR(8),
    3 YR PIC '99',
    3 WOL PIC '99',
    3 F3 CHAR(3),
    3 OCCP PIC '99',
    3 F4 CHAR(12),
    3 AG PIC '999',
    3 F5 CHAR(23);
DCL AGG DEF T_R POS(55) CHAR(7);
DCL TAPEOUT FILE RECORD OUTPUT;
DCL 1 TA_R,
    3 PY0 PIC '99',
    3 PYR PIC '99',
    3 PJUL PIC '999',
    3 PCAN PIC '99',
    3 TAA(40) PIC '(8)9';
DCL YEAR(10) PIC '99' INIT(82,81,80,79,78,77,76,75,74,73);
DCL AGES(121) PIC '999' INIT(T,T,0,1,2,3,4,TOT,5,6,7,8,9,TOT,
    10,11,12,13,14,TOT,15,16,17,18,19,TOT,
    20,21,22,23,24,TOT,25,26,27,28,29,TOT,
    30,31,32,33,34,TOT,35,36,37,38,39,TOT,
    40,41,42,43,44,TOT,45,46,47,48,49,TOT,
    50,51,52,53,54,TOT,55,56,57,58,59,TOT,
    60,61,62,63,64,TOT,65,66,67,68,69,TOT,
    70,71,72,73,74,TOT,75,76,77,78,79,TOT,
SVD00010
SVD00020
SVD00030
SVD00040
SVD00050
SVD00060
SVD00070
SVD00080
SVD00090
SVD00100
SVD00110
SVD00120
SVD00130
SVD00140
SVD00150
SVD00160
SVD00170
SVD00180
SVD00190
SVD00200
SVD00210
SVD00220
SVD00230
SVD00240
SVD00250
SVD00260
SVD00270
SVD00280
SVD00290
SVD00300
SVD00310
SVD00320
SVD00330
SVD00340
SVD00350
SVD00360
SVD00370
SVD00380
SVD00390
SVD00400
SVD00410
SVD00420
SVD00430
SVD00440
SVD00450
SVD00460
SVD00470
SVD00480
SVD00490
SVD00500
SVD00510
SVD00520
SVD00530
SVD00540
SVD00550

```

```

80,81,82,83,84,TOT,85,86,87,88,89,TOT, SVD00560
90,91,92,93,94,TOT,95,96,97,98,99); SVD00570
DCL AGE9(22) PIC '999' INIT(0,1,5,10,15,20,25,30,35,40, SVD00580
45,50,55,60,65,70,75,80,85,90,95,100); SVD00590
DCL DOD(14) PIC '99' INIT(0,11,21,22,23,31,32,33,34,35,36,37,38,39); SVD00600
DCL DEC(11) PIC '99' INIT(0,1,2,3,4,5,6,7,11,12,13); SVD00610
DCL TAB7(10,13,3,15) FIXED DEC(8); SVD00620
DCL TAB8(10,123,15,2) FIXED DEC(8); SVD00630
DCL TAB9(10,3,24,11) FIXED DEC(8); SVD00640
DCL END_RECORDS CHAR(3); SVD00650
/* SUB PROCEDURE */ SVD00660
SET_RTN: PROCEDURE; SVD00670
DO N =1 TO 10; SVD00680
IF YR = YEAR(N) THEN DO; I=N; SVD00690
GOTO SET1; SVD00700
END; SVD00710
END; SVD00720
GOTO SET6; SVD00730
SET1: SVD00740
DO N=1 TO 12; SVD00750
IF WOL = N THEN DO; J7=N+1; SVD00760
GOTO SET2; SVD00770
END; SVD00780
END; SVD00790
J7=13; SVD00800
SET2: SVD00810
K7=SEX+1; SVD00820
DO N=2 TO 14; SVD00830
IF DO = DOD(N) THEN DO; L7=N; SVD00840
GO TO SET3; SVD00850
END; SVD00860
END; SVD00870
L7=15; SVD00880
SET3: SVD00890
DO N = 3 TO 115; SVD00900
IF AG = AGE8(N) THEN DO; J8=N; SVD00910
GOTO SET4; SVD00920
END; SVD00930
END; SVD00940
J8=116; SVD00950
SET4: SVD00960
K8=L7; SVD00970
L8=SEX; SVD00980
J9=K7; SVD00990
DO N=2 TO 21; SVD01000
IF AG < AGE9(N) THEN DO; K9=N; SVD01010
GO TO SET5; SVD01020
END; SVD01030
END; SVD01040
K9=22; SVD01050
SET5: SVD01060
DO N=2 TO 11; SVD01070
IF OCCP=OCC(N) THEN DO; L9=N; SVD01080
GOTO SET5; SVD01090
END; SVD01100

```



```

END;
L9=11;
SET55:
IF L9 = 11 & AG < 15 THEN L9 =9;
TAB7(I,J7,K7,L7) = TAB7(I,J7,K7,L7)+1;
TAB7(I,I,K7,L7) = TAB7(I,I,K7,L7)+1;
TAB8(I,J8,K8,L8) = TAB8(I,J8,K8,L8)+1;
TAB8(I,I,K8,L8) = TAB8(I,I,K8,L8)+1;
TAB9(I,J9,K9,L9) = TAB9(I,J9,K9,L9)+1;
TAB9(I,I,K9,L9) = TAB9(I,I,K9,L9)+1;
SET6:
END SET_RTN;
SUM_RTN: PROC;
TAB7(*,*,1,*)=TAB7(*,*,2,*) + TAB7(*,*,3,*);
DO I=2 TO 15;
    TAB7(*,*,*,1)=TAB7(*,*,*,1)+TAB7(*,*,*,I);
END;
DO I=2 TO 110 BY 6;
    DO J=I+1 TO I+5;
        TAB8(*,I,*,*)=TAB8(*,I,*,*)+TAB8(*,J,*,*);
    END;
END;
DO I =2 TO 15;
    TAB8(*,*,1,1)=TAB8(*,*,1,1) + TAB8(*,*,I,1);
    TAB8(*,*,1,2)=TAB8(*,*,1,2) + TAB8(*,*,I,2);
END;
DO I=1 TO 3;
    DO J=2 TO 22;
        TAB9(*,I,1,*) = TAB9(*,I,1,*)+TAB9(*,I,J,*);
    END;
END;
DO I=2 TO 11;
    TAB9(*,*,*,1) = TAB9(*,*,*,1)+TAB9(*,*,*,I);
END;
END SUM_RTN;
WRITE_RTN: PROCEDURE ;
CALL SUBWR (TAB7,7,10,13,3,15);
N=0;
DO I=1 TO 10;
    DO J=1 TO 117;
        PYO=8;
        PYR=I;
        PJUL=J;
        DO K=1 TO 15;
            DO L=1 TO 2;
                PCAN=K;
                N=N+1;
                TAA(N)=TAB8(I,J,K,L);
            END;
        END;
        WRITE FILE(TAPEOUT) FROM (TA_R);
        N=0;
    END;
END;
CALL SUBWR(TAB9,9,10,3,23,11);
END WRITE_RTN;

```

SVD01110  
SVD01120  
SVD01130  
SVD01140  
SVD01150  
SVD01160  
SVD01170  
SVD01180  
SVD01190  
SVD01200  
SVD01210  
SVD01220  
SVD01230  
SVD01240  
SVD01250  
SVD01260  
SVD01270  
SVD01280  
SVD01290  
SVD01300  
SVD01310  
SVD01320  
SVD01330  
SVD01340  
SVD01350  
SVD01360  
SVD01370  
SVD01380  
SVD01390  
SVD01400  
SVD01410  
SVD01420  
SVD01430  
SVD01440  
SVD01450  
SVD01460  
SVD01470  
SVD01480  
SVD01490  
SVD01500  
SVD01510  
SVD01520  
SVD01530  
SVD01540  
SVD01550  
SVD01560  
SVD01570  
SVD01580  
SVD01590  
SVD01600  
SVD01610  
SVD01620  
SVD01630  
SVD01640  
SVD01650

```

SUBWR: PROC(ARRAY,PP,II,JJ,KK,LL);
DCL ARRAY(*,*,*,*) FIXED DEC(8);
DO I= 1 TO II;
  NN=0;
  DO J=1 TO JJ;
    DO K=1 TO KK;
      NN=NN+1;
      PYO=PP;
      PYR=I;
      PJUL=NN;
      PCAN=K;
      DO N=1 TO LL;
        TAA(N) = ARRAY(I,J,K,N);
      END;
      WRITE FILE(TAPEOUT) FROM(TA_LR);
    END;
  END;
END SUBWR;
END CHOI;
/*
//GO.TAPEIN DD UNIT=TAPE,DISP=(OLD,KEEP),VOL=SER=CHOI13,
//          DSN=CHOI1, LABEL=(2, BLP),
//          DCB=(LRECL=80,RECFM=FB,BLKSIZE=3200,EROPT=SKP)
//GO.TAPEOUT DD UNIT=TAPE,DISP=(NEW,KEEP),DSN=CHOI1D,
//           DCB=(LRECL=329,RECFM=FB,BLKSIZE=3290,EROPT=SKP)
//GO.SYSPRINT DD SYSOUT=A
//GO.SYSOUT DD SYSOUT=A
//
SVD01660
SVD01670
SVD01680
SVD01690
SVD01700
SVD01710
SVD01720
SVD01730
SVD01740
SVD01750
SVD01760
SVD01770
SVD01780
SVD01790
SVD01800
SVD01810
SVD01820
SVD01830
SVD01840
SVD01850
SVD01860
SVD01870
SVD01880
SVD01890
SVD01900
SVD01910
SVD01920
SVD01930
SVD01940

```

```

//SVDWDL14 JOE TYPRUN=HOLD
//* DEATH TABLE SUMMARY TO TAPE SUM ADD UP *****
// EXEC PLIFCLG
//PLI.SYSIN DD *
* PROCESS GS,NEST,OPT(TIME);
CHOI: PROC OPTIONS(MAIN);
    TAB7=0;
    TAB8=0;
    TAB9=0;
    OPEN FILE(TAPEIN1) INPUT,
        FILE(TAPEIN2) INPUT,
        FILE(TAPEOUT) OUTPUT,
        FILE(SYSPRINT) LINESIZE(132) PAGESIZE(60);
    ON ENDFILE(TAPEIN1) END1_REC='YES';
    ON ENDFILE(TAPEIN2) END2_REC='YES';
    END1_REC = 'NO ';
    END2_REC = 'NO ';
/* TAPE READ PART */
    READ FILE(TAPEIN1) INTO(T_R);
    DO WHILE(END1_REC = 'NO ');
        IF GU = 7 THEN CALL SET7_R ;
        ELSE
            IF GU = 8 THEN CALL SET8_R ;
            ELSE CALL SET9_R ;
            READ FILE(TAPEIN1) INTO(T_R);
        END;
        READ FILE(TAPEIN2) INTO(TI_R);
        DO WHILE(END2_REC = 'NO ');
            IF KU = 7 THEN CALL SET7_RTN;
            ELSE
                IF KU = 8 THEN CALL SET8_RTN;
                ELSE CALL SET9_RTN;
            READ FILE(TAPEIN2) INTO(TI_R);
        END;
        CALL WRITE_RTN;
        CLOSE FILE(TAPEIN1),FILE(TAPEIN2),FILE(TAPEOUT);
/* DECLARE PART */
DCL TAPEIN1 FILE RECORD INPUT;
DCL 1 T_R,
    3 GU PIC '99',
    3 GU1 PIC '99',
    3 GU2 PIC '999',
    3 GU3 PIC '99',
    3 AA(40) PIC '(8)9';
DCL TAPEIN2 FILE RECORD INPUT;
DCL 1 TI_R,
    3 KU PIC '99',
    3 KU1 PIC '99',
    3 KU2 PIC '999',
    3 KU3 PIC '99',
    3 BB(40) PIC '(8)9';
DCL TAPEOUT FILE RECORD OUTPUT;
DCL 1 TA_R,
    3 GUA PIC '99',
    3 GUB PIC '99',

```

SVD00010  
SVD00020  
SVD00030  
SVD00040  
SVD00050  
SVD00060  
SVD00070  
SVD00080  
SVD00090  
SVD00100  
SVD00110  
SVD00120  
SVD00130  
SVD00140  
SVD00150  
SVD00160  
SVD00170  
SVD00180  
SVD00190  
SVD00200  
SVD00210  
SVD00220  
SVD00230  
SVD00240  
SVD00250  
SVD00260  
SVD00270  
SVD00280  
SVD00290  
SVD00300  
SVD00310  
SVD00320  
SVD00330  
SVD00340  
SVD00350  
SVD00360  
SVD00370  
SVD00380  
SVD00390  
SVD00400  
SVD00410  
SVD00420  
SVD00430  
SVD00440  
SVD00450  
SVD00460  
SVD00470  
SVD00480  
SVD00490  
SVD00500  
SVD00510  
SVD00520  
SVD00530  
SVD00540  
SVD00550

```

3 GUC PIC '999', SVD00560
3 GUD PIC '99', SVD00570
3 CC(40) PIC '(8)9', SVD00580
DCL TAB7(10,39,15) FIXED DEC(8); SVD00590
DCL TAB8(10,123,30) FIXED DEC(8); SVD00600
DCL TAB9(10,72,11) FIXED DEC(8); SVD00610
DCL END1_REC CHAR(3); SVD00620
DCL END2_REC CHAR(3); SVD00630
/**** SUB PROCEDURE ****/ SVD00640
SET7_R: PROC; SVD00650
IF GU1 = 10 THEN GOTO SET7_E; SVD00660
I=GU1+1; SVD00670
J=GU2; SVD00680
DO K=1 TO 15; SVD00690
TAB7(I,J,K) = AA(K); SVD00700
END; SVD00710
SET7_E: END SET7_R; SVD00720
SET8_R: PROC; SVD00730
IF GU1=10 THEN GOTO SET8_E; SVD00740
I=GU1+1; SVD00750
IF GU2 < 117 THEN J = GU2; SVD00760
ELSE J= 117; SVD00770
DO K=1 TO 30; SVD00780
TAB8(I,J,K) = AA(K); SVD00790
END; SVD00800
SET8_E: END SET8_R; SVD00810
SET9_R: PROC; SVD00820
IF GU1 = 10 THEN GOTO SET9_E; SVD00830
I=GU1+1; SVD00840
IF GU2 < 23 THEN J=GU2; SVD00850
ELSE IF GU2 < 47 THEN J=GU2 -1; SVD00860
ELSE IF GU2 < 71 THEN J=GU2 -2; SVD00870
ELSE J=GU2 -3; SVD00880
DO K=1 TO 11; SVD00890
TAB9(I,J,K) = AA(K); SVD00900
END; SVD00910
SET9_E: END SET9_R; SVD00920
SET7_RTN: PROC; SVD00930
I=KU1; SVD00940
J=KU2; SVD00950
DO K=1 TO 15; SVD00960
TAB7(I,J,K)=TAB7(I,J,K) + BB(K); SVD00970
END; SVD00980
END SET7_RTN; SVD00990
SET8_RTN: PROC; SVD01000
I=KU1; SVD01010
J=KU2; SVD01020
DO K=1 TO 30; SVD01030
TAB8(I,J,K)=TAB8(I,J,K) + BB(K); SVD01040
END; SVD01050
END SET8_RTN; SVD01060
SET9_RTN: PROC; SVD01070
I=KU1; SVD01080
J=KU2; SVD01090
DO K=1 TO 11; SVD01100

```

```

                TAB9(I,J,K)= TAB9(I,J,K) + BB(K);
            END;
        END SET9_RTN;
    WRITE_RTN: PROC;
        CALL SUBWR(TAB7,10,39,15,7) ;
        CALL SUBWR(TAB8,10,117,30,8);
        CALL SUBWR(TAB9,10,69,11,9);
    END WRITE_RTN;
    SUBWR: PROC(ARRAY,II,JJ,KK,PP);
    DCL  ARRAY(*,*,*) FIXED DEC(8);
        DO I=1 TO II;
            DO J=1 TO JJ;
                GUA=PP;
                GUB=I;
                GUC=J;
                DO K=1 TO KK;
                    CC(K) = ARRAY(I,J,K);
                END;
                WRITE FILE(TAPEOUT) FROM(TA_R);
            END;
        END;
    END SUBWR;
END CHOI;
/*
//GO.TAPEIN1 DD DSN=SVDWED,UNIT=TAPE,DISP=(OLD,KEEP),VOL=SER=AAAAAA,
//            DCB=(RECFM=FB,LRECL=329,BLKSIZE=3290,EROPT=SKP),
//            LABEL=(2,BLP)
//GO.TAPEIN2 DD DSN=SVDWET,UNIT=TAPE,DISP=(OLD,KEEP),VOL=SER=AA2376,
//            DCB=(RECFM=FB,LRECL=329,BLKSIZE=3290,EROPT=SKP),
//            LABEL=(2,BLP)
//GO.TAPEOUT DD DSN=SVDWMT,UNIT=TAPE,DISP=(NEW,KEEP),
//            DCB=(RECFM=FB,LRECL=329,BLKSIZE=3290,EROPT=SKP)
//GO.SYSPRINT DD SYSOUT=A
//GO.SYSOUT DD SYSOUT=A
//
SVD01110
SVD01120
SVD01130
SVD01140
SVD01150
SVD01160
SVD01170
SVD01180
SVD01190
SVD01200
SVD01210
SVD01220
SVD01230
SVD01240
SVD01250
SVD01260
SVD01270
SVD01280
SVD01290
SVD01300
SVD01310
SVD01320
SVD01330
SVD01340
SVD01350
SVD01360
SVD01370
SVD01380
SVD01390
SVD01400
SVD01410
SVD01420
SVD01430
SVD01440
SVD01450

```

```

//SVDWTE14 JOB TYPRUN=HOLD
//TABSM EXEC COBUCLG,PARM='NOSOURCE'
//* ** TABLE 7 8 9 EDITING **
//COB.SYSIN DD *
  ID DIVISION.
  PROGRAM-ID. TED789.
  ENVIRONMENT DIVISION.
  CONFIGURATION SECTION.
  SPECIAL-NAMES.
    C01 IS CHOI.
    CSP IS DOUB.
  INPUT-OUTPUT SECTION.
    SELECT T-F ASSIGN UT-S-TAPEIN.
    SELECT P-F ASSIGN UT-S-PROUT.
  DATA DIVISION.
  FILE SECTION.
    FD T-F LABEL RECORD STANDARD
      BLOCK CONTAINS 10 RECORDS.
    01 T-R.
      03 GU PIC 99.
      03 GU1 PIC 99.
      03 GU2 PIC 999.
      03 GU3 PIC 99.
      03 AA OCCURS 40 PIC 9(8).
    FD P-F LABEL RECORD OMITTED.
    01 P-R PIC X(132).
  WORKING-STORAGE SECTION.
    77 I PIC 999 VALUE 0.
    77 J PIC 999 VALUE 0.
    77 K PIC 999 VALUE 0.
    77 L PIC 999 VALUE 1.
    77 M PIC 999 VALUE 2.
    77 N PIC 999 VALUE 3.
    77 P PIC 999 VALUE 0.
    77 J1 PIC 9(3) VALUE 0.
    77 LL PIC 9(3) VALUE 2.
    77 D1 PIC 99 VALUE 0.
    77 SW7 PIC 9 VALUE 0.
    77 SW9 PIC 9 VALUE 0.
    77 B1 PIC 99 VALUE 1.
    77 BIGY0 PIC 99 VALUE 0.
    77 SW8 PIC 9 VALUE 0.
    01 LIST7.
      03 F PIC X(15) VALUE SPACE.
      03 WHL7 PIC Z(7)9.
      03 P7 OCCURS 12 PIC Z(7)9.
      03 P71 PIC Z(6)9.
      03 P72 PIC Z(5)9.
    01 LIST81.
      03 F PIC X(12) VALUE SPACE.
      03 WHL8 PIC Z(8)9.
      03 WHLM PIC Z(8)9.
      03 WHLF PIC Z(8)9.
      03 P81 OCCURS 10 PIC Z(7)9.
    01 LIST82.

```

SVD00010  
SVD00020  
SVD00030  
SVD00040  
SVD00050  
SVD00060  
SVD00070  
SVD00080  
SVD00090  
SVD00100  
SVD00110  
SVD00120  
SVD00130  
SVD00140  
SVD00150  
SVD00160  
SVD00170  
SVD00180  
SVD00190  
SVD00200  
SVD00210  
SVD00220  
SVD00230  
SVD00240  
SVD00250  
SVD00260  
SVD00270  
SVD00280  
SVD00290  
SVD00300  
SVD00310  
SVD00320  
SVD00330  
SVD00340  
SVD00350  
SVD00360  
SVD00370  
SVD00380  
SVD00390  
SVD00400  
SVD00410  
SVD00420  
SVD00430  
SVD00440  
SVD00450  
SVD00460  
SVD00470  
SVD00480  
SVD00490  
SVD00500  
SVD00510  
SVD00520  
SVD00530  
SVD00540  
SVD00550

03	P821	OCCURS 4 PIC Z(6)9.	SVD00560
03	P822	OCCURS 2.	SVD00570
	05 P8222	PIC Z(7)9.	SVD00580
	05 P8223	PIC Z(6)9.	SVD00590
03	P833	OCCURS 4 PIC Z(7)9.	SVD00600
03	P844	OCCURS 2 PIC Z(6)9.	SVD00610
03	P855	OCCURS 2 PIC Z(5)9.	SVD00620
01	LIST9.		SVD00630
	03 F	PIC X(22) VALUE SPACE.	SVD00640
	03 P9	OCCURS 11 PIC Z(9)9.	SVD00650
01	TABL8.		SVD00660
	05 TAB	OCCURS 123.	SVD00670
	07 T8	OCCURS 31 PIC 9(8).	SVD00680
	PROCEDURE DIVISION.		SVD00690
	OPEN INPUT T-F.		SVD00700
	OPEN OUTPUT P-F.		SVD00710
	MOVE SPACE TO P-R.		SVD00720
	BUSAN.		SVD00730
	WRITE P-R AFTER ADVANCING CHOI.		SVD00740
	MOVE SPACE TO P-R.		SVD00750
	WRITE P-R AFTER ADVANCING 3.		SVD00760
	MOVE SPACE TO P-R.		SVD00770
	WRITE P-R AFTER ADVANCING 3.		SVD00780
	MOVE SPACE TO P-R.		SVD00790
	WRITE P-R AFTER ADVANCING 3.		SVD00800
	MOVE SPACE TO P-R.		SVD00810
	BUSAN1.		SVD00820
	WRITE P-R AFTER ADVANCING 2.		SVD00830
	MOVE SPACE TO P-R.		SVD00840
	BUSAN2.		SVD00850
	WRITE P-R AFTER ADVANCING 3.		SVD00860
	MOVE SPACE TO P-R.		SVD00870
	READ-RTN.		SVD00880
	READ T-F AT END GO TO END-RTN.		SVD00890
	COMPUTE D1 = GU - 6.		SVD00900
	GO TO 7T 100 300 DEPENDING ON D1.		SVD00910
	7T.		SVD00920
	IF SW7 = 1 GO TO PRT7.		SVD00930
	MOVE GU1 TO BIGYO.		SVD00940
	MOVE 1 TO SW7.		SVD00950
	GO TO PRT77.		SVD00960
	PRT7.		SVD00970
	IF GU1 NOT = BIGYO PERFORM BUSAN THRU BUSAN2		SVD00980
	MOVE GU1 TO BIGYO		SVD00990
	MOVE 1 TO L.		SVD01000
	PRT77.		SVD01010
	MOVE AA(1) TO WHL7.		SVD01020
	PERFORM PR7 VARYING I FROM 2 BY 1 UNTIL I > 13.		SVD01030
	MOVE AA(14) TO P71.		SVD01040
	MOVE AA(15) TO P72.		SVD01050
	IF GU2 = L WRITE P-R FROM LIST7 AFTER ADVANCING 3		SVD01060
	WRITE P-R FROM LIST7 AFTER ADVANCING DOUB		SVD01070
	MOVE SPACE TO P-R		SVD01080
	COMPUTE L = L + 3		SVD01090
	GO TO READ-RTN.		SVD01100

WRITE P-R FROM LIST7 AFTER ADVANCING 2.	SVD01110
MOVE SPACE TO P-R.	SVD01120
GO TO READ-RTN.	SVD01130
PR7.	SVD01140
COMPUTE J = I - 1.	SVD01150
MOVE AA (I) TO P7 (J).	SVD01160
100.	SVD01170
IF GU1 NOT = B1 GO TO BPRT-RTN.	SVD01180
RT. MOVE GU2 TO I.	SVD01190
COMPUTE T8 (I 1) = AA (1) + AA (2).	SVD01200
PERFORM PR8 VARYING K FROM 1 BY 1 UNTIL K > 30.	SVD01210
GO TO READ-RTN.	SVD01220
PR8.	SVD01230
COMPUTE P = K + 1.	SVD01240
MOVE AA (K) TO T8 (I P).	SVD01250
BPRT-RTN.	SVD01260
PERFORM BUSAN THRU BUSAN1.	SVD01270
PERFORM PT8 THRU ET VARYING I FROM 1 BY 1 UNTIL I > 61.	SVD01280
MOVE 2 TO M.	SVD01290
MOVE 3 TO N.	SVD01300
PERFORM BUSAN THRU BUSAN1.	SVD01310
PERFORM PTE THRU EX VARYING I FROM 1 BY 1 UNTIL I > 61.	SVD01320
MOVE 62 TO M.	SVD01330
MOVE 63 TO N.	SVD01340
PERFORM BUSAN THRU BUSAN1.	SVD01350
PERFORM PTG THRU EJ VARYING I FROM 62 BY 1 UNTIL I > 117.	SVD01360
MOVE 62 TO M.	SVD01370
MOVE 63 TO N.	SVD01380
PERFORM BUSAN THRU BUSAN1.	SVD01390
PERFORM PTT THRU EK VARYING I FROM 62 BY 1 UNTIL I > 117.	SVD01400
BB. MOVE 2 TO M.	SVD01410
MOVE 3 TO N.	SVD01420
MOVE GU1 TO B1.	SVD01430
GO TO RT.	SVD01440
PT8.	SVD01450
MOVE T8 (I 1) TO WHL8.	SVD01460
MOVE T8 (I 2) TO WHLM.	SVD01470
MOVE T8 (I 3) TO WHLF.	SVD01480
PERFORM P88 VARYING J FROM 4 BY 1 UNTIL J > 13.	SVD01490
IF I = 1 WRITE P-R FROM LIST81 AFTER ADVANCING 3	SVD01500
WRITE P-R FROM LIST81 AFTER ADVANCING DOUB	SVD01510
MOVE SPACE TO P-R	SVD01520
GO TO ET.	SVD01530
IF I = M WRITE P-R FROM LIST81 AFTER ADVANCING 3	SVD01540
WRITE P-R FROM LIST81 AFTER ADVANCING DOUB	SVD01550
MOVE SPACE TO P-R	SVD01560
COMPUTE M = M + 6	SVD01570
GO TO ET.	SVD01580
IF I = N WRITE P-R FROM LIST81 AFTER ADVANCING 2	SVD01590
MOVE SPACE TO P-R	SVD01600
COMPUTE N = N + 6	SVD01610
GO TO ET.	SVD01620
WRITE P-R FROM LIST81 AFTER ADVANCING 1.	SVD01630
MOVE SPACE TO P-R.	SVD01640
ET. EXIT.	SVD01650



```

P88. COMPUTE P = J - 3. SVD01660
    MOVE T8 (I J) TO P81 (P). SVD01670
PTE. SVD01680
    PERFORM T821 VARYING J FROM 14 BY 1 UNTIL J > 17. SVD01690
    MOVE T8(I 18) TO P8222(1). SVD01700
    MOVE T8(I 19) TO P8223(1). SVD01710
    MOVE T8(I 20) TO P8222(2). SVD01720
    MOVE T8(I 21) TO P8223(2). SVD01730
    PERFORM T833 VARYING J FROM 22 BY 1 UNTIL J > 27. SVD01740
    MOVE T8(I 28) TO P844(1). SVD01750
    MOVE T8(I 29) TO P844(2). SVD01760
    MOVE T8(I 30) TO P855(1). SVD01770
    MOVE T8(I 31) TO P855(2). SVD01780
    IF I = 1 WRITE P-R FROM LIST82 AFTER ADVANCING 3 SVD01790
        WRITE P-R FROM LIST82 AFTER ADVANCING DOUB SVD01800
        MOVE SPACE TO P-R SVD01810
        GO TO EX. SVD01820
    IF I = M WRITE P-R FROM LIST82 AFTER ADVANCING 3 SVD01830
        WRITE P-R FROM LIST82 AFTER ADVANCING DOUB SVD01840
        MOVE SPACE TO P-R SVD01850
        COMPUTE M = M + 6 SVD01860
        GO TO EX. SVD01870
    IF I = N WRITE P-R FROM LIST82 AFTER ADVANCING 2 SVD01880
        MOVE SPACE TO P-R SVD01890
        COMPUTE N = N + 6 SVD01900
        GO TO EX. SVD01910
    WRITE P-R FROM LIST82 AFTER ADVANCING 1. SVD01920
    MOVE SPACE TO P-R. SVD01930
EX. EXIT. SVD01940
T821. COMPUTE P = J - 13. SVD01950
    MOVE T8 (I J) TO P821 (P). SVD01960
T833. COMPUTE P = J - 21. SVD01970
    MOVE T8 (I J) TO P833 (P). SVD01980
PTG. SVD01990
    MOVE T8 (I 1) TO WHL8. SVD02000
    MOVE T8 (I 2) TO WHLM. SVD02010
    MOVE T8 (I 3) TO WHLF. SVD02020
    PERFORM R88 VARYING J FROM 4 BY 1 UNTIL J > 13. SVD02030
    IF I = 117 WRITE P-R FROM LIST81 AFTER ADVANCING 3 SVD02040
        MOVE SPACE TO P-R SVD02050
        GO TO EJ. SVD02060
    IF I = M WRITE P-R FROM LIST81 AFTER ADVANCING 3 SVD02070
        WRITE P-R FROM LIST81 AFTER ADVANCING DOUB SVD02080
        MOVE SPACE TO P-R COMPUTE M = M + 6 SVD02090
        GO TO EJ. SVD02100
    IF I = N WRITE P-R FROM LIST81 AFTER ADVANCING 2 SVD02110
        MOVE SPACE TO P-R SVD02120
        COMPUTE N = N + 6 SVD02130
        GO TO EJ. SVD02140
    WRITE P-R FROM LIST81 AFTER ADVANCING 1. SVD02150
    MOVE SPACE TO P-R. SVD02160
EJ. EXIT. SVD02170
R88. SVD02180
    COMPUTE P = J - 3. SVD02190
    MOVE T8 (I J) TO P81(P). SVD02200

```

PTT.		SVD02210
PERFORM N821	VARYING J FROM 14 BY 1 UNTIL J > 17.	SVD02220
MOVE T8(I 18) TO P8222(1).		SVD02230
MOVE T8(I 19) TO P8223(1).		SVD02240
MOVE T8(I 20) TO P8222(2).		SVD02250
MOVE T8(I 21) TO P8223(2).		SVD02260
PERFORM N833	VARYING J FROM 22 BY 1 UNTIL J > 27.	SVD02270
MOVE T8(I 28) TO P844(1).		SVD02280
MOVE T8(I 29) TO P844(2).		SVD02290
MOVE T8(I 30) TO P855(1).		SVD02300
MOVE T8(I 31) TO P855(2).		SVD02310
IF I = 117	WRITE P-R FROM LIST82 AFTER ADVANCING 3	SVD02320
	MOVE SPACE TO P-R	SVD02330
	GO TO EK.	SVD02340
IF I = M	WRITE P-R FROM LIST82 AFTER ADVANCING 3	SVD02350
	WRITE P-R FROM LIST82 AFTER ADVANCING DOUB	SVD02360
	MOVE SPACE TO P-R	SVD02370
	COMPUTE M = M + 6	SVD02380
	GO TO EK.	SVD02390
IF I = N	WRITE P-R FROM LIST82 AFTER ADVANCING 2	SVD02400
	MOVE SPACE TO P-R	SVD02410
	COMPUTE N = N + 6	SVD02420
	GO TO EK.	SVD02430
	WRITE P-R FROM LIST82 AFTER ADVANCING 1.	SVD02440
	MOVE SPACE TO P-R.	SVD02450
EK. EXIT.		SVD02460
N821. COMPUTE P = J - 13.		SVD02470
	MOVE T8 (I J) TO P821 (P).	SVD02480
N833. COMPUTE P = J - 21.		SVD02490
	MOVE T8 (I J) TO P833 (P).	SVD02500
300.		SVD02510
	IF SW9 = 1 GO TO 9PRT.	SVD02520
	PERFORM 8PRT-RTN.	SVD02530
	MOVE GU1 TO BIGYO.	SVD02540
	PERFORM BUSAN.	SVD02550
	MOVE 1 TO SW9.	SVD02560
	MOVE 1 TO M.	SVD02570
	MOVE 2 TO N.	SVD02580
	MOVE 23 TO P.	SVD02590
9PRT.		SVD02600
	IF GU1 NOT = BIGYO MOVE GU1 TO BIGYO	SVD02610
	PERFORM BUSAN	SVD02620
	MOVE 1 TO M	SVD02630
	MOVE 2 TO N	SVD02640
	MOVE 23 TO P.	SVD02650
PERFORM PR9	VARYING I FROM 1 BY 1 UNTIL I > 11.	SVD02660
IF GU2 = P	WRITE P-R FROM LIST9 AFTER ADVANCING 2	SVD02670
	MOVE SPACE TO P-R	SVD02680
	COMPUTE P = P + 23	SVD02690
	COMPUTE N = N - 1	SVD02700
	GO TO READ-RTN.	SVD02710
IF GU2 = M	WRITE P-R FROM LIST9 AFTER ADVANCING 3	SVD02720
	WRITE P-R FROM LIST9 AFTER ADVANCING DOUB	SVD02730
	MOVE SPACE TO P-R	SVD02740
	COMPUTE M = M + 23	SVD02750

```

                                GO TO READ-RTN.
IF GU2 = N      WRITE P-R FROM LIST9 AFTER ADVANCING 2
                MOVE SPACE TO P-R
                COMPUTE N = N + 4
                GO TO READ-RTN.
WRITE P-R FROM LIST9 AFTER ADVANCING 1.
MOVE SPACE TO P-R.
GO TO READ-RTN.
PR9.
  MOVE AA (I) TO P9 (I).
END-RTN.
CLOSE T-F P-F.
STOP RUN.
/*
//GD.TAPEIN DD UNIT=TAPE,DISP=(OLD,KEEP),DSN=VBSM13,VOL=SER=VB1111,
//          LABEL=(2,BLF)
//SYSOUT   DD SYSOUT=A
//GD.PROUT DD SYSOUT=A
//SYSDBOUT DD SYSOUT=A
//
SVD02760
SVD02770
SVD02780
SVD02790
SVD02800
SVD02810
SVD02820
SVD02830
SVD02840
SVD02850
SVD02860
SVD02870
SVD02880
SVD02890
SVD02900
SVD02910
SVD02920
SVD02930
SVD02940
SVD02950
```

//SVMWTT14 JOB TYPRUN=HOLD	SVM00010
//TTT EXEC COBUCLG	SVM00020
//COB.SYSIN DD *	SVM00030
ID DIVISION.	SVM00040
PROGRAM-ID. TTT.	SVM00050
ENVIRONMENT DIVISION.	SVM00060
INPUT-OUTPUT SECTION.	SVM00070
FILE-CONTROL.	SVM00080
SELECT T-F ASSIGN UT-S-TAPEIN.	SVM00090
SELECT TA-F ASSIGN UT-S-TAPEOUT.	SVM00100
DATA DIVISION.	SVM00110
FILE SECTION.	SVM00120
FD T-F LABEL RECORD OMITTED.	SVM00130
01 T-R.	SVM00140
03 AA PIC X(13).	SVM00150
03 BB PIC X(38).	SVM00160
03 CC PIC X(3).	SVM00170
03 DD REDEFINES CC PIC 999.	SVM00180
03 F PIC X(26).	SVM00190
FD TA-F BLOCK CONTAINS 40 RECORDS	SVM00200
LABEL RECORD STANDARD.	SVM00210
01 TA-R.	SVM00220
02 TAA.	SVM00230
03 TA PIC X(13).	SVM00240
03 TC PIC 999.	SVM00250
02 TBB.	SVM00260
03 TB PIC X(38).	SVM00270
03 F PIC X(26).	SVM00280
WORKING-STORAGE SECTION.	SVM00290
77 ECNT PIC 9(7) VALUE 0.	SVM00300
77 OCNT PIC 9(7) VALUE 0.	SVM00310
PROCEDURE DIVISION.	SVM00320
OPEN INPUT T-F.	SVM00330
OPEN OUTPUT TA-F.	SVM00340
MOVE SPACE TO TA-R.	SVM00350
READ-RTN.	SVM00360
READ T-F AT END GO TO END-RTN.	SVM00370
ADD 1 TO ECNT.	SVM00380
EXAMINE CC TALLYING ALL SPACE REPLACING BY 0.	SVM00390
IF DD NOT NUMERIC GO TO MOVE-RTN.	SVM00400
COMPUTE TC = DD / 10 ** TALLY.	SVM00410
MOVE-RTN.	SVM00420
MOVE AA TO TA.	SVM00430
MOVE BB TO TB.	SVM00440
WRITE TA-R.	SVM00450
MOVE SPACE TO TA-R.	SVM00460
ADD 1 TO OCNT.	SVM00470
GO TO READ-RTN.	SVM00480
END-RTN.	SVM00490
DISPLAY ' ECNT = ' ECNT.	SVM00500
DISPLAY ' OCNT = ' OCNT.	SVM00510
CLOSE T-F TA-F.	SVM00520
OPEN INPUT TA-F.	SVM00530
RD-RTN.	SVM00540
READ TA-F AT END CLOSE TA-F STOP RUN.	SVM00550

FILE: SVMWTT14 RJTT A1 VM/SP RELEASE 2.1 EXPRESS PUT8302+ SLU203

```
        DISPLAY " " TAA.                                SVM00560
    EX.  PERFORM RD-RTN 1000 TIMES.                       SVM00570
        CLOSE TA-F.                                     SVM00580
        STOP RUN.                                       SVM00590
/*                                                  SVM00600
//GO.TAPEIN DD DSN=VBCHOI,UNIT=TAPE,DISP=OLD,VOL=SER=000001, SVM00610
//          DCB=EROPT=SKP,LABEL=(,NL)                  SVM00620
//          DD DSN=HB,UNIT=AFF=TAPEIN,LABEL=(,NL),VOL=SER=000002 SVM00630
//          DD DSN=HB,UNIT=AFF=TAPEIN,LABEL=(,NL),VOL=SER=000003 SVM00640
//GO.TAPEOUT DD DSN=CHOI13,UNIT=TAPE,DISP=NEW,VOL=SER=SCRATCH SVM00650
//SYSOUT DD SYSOUT=A                                  SVM00660
//                                                  SVM00670
```

```

//SVMWMS14 JOB TYPRUN=HOLD
/** MARRIAGE TABLE SUMMARY ***
// EXEC PLIFCLG
//PLI.SYSIN DD *
* PROCESS GS,NEST,OPT(TIME);
  CHOI: PROC OPTIONS(MAIN);
    TAB10=0;
    TAB11=0;
    TAB12=0;
    TAB13=0;
    OPEN FILE(TAPEIN) INPUT,
    FILE(TAPEOUT) OUTPUT,
    FILE(SYSPRINT) LINESIZE(132) PAGESIZE(60);
    ON ENDFILE(TAPEIN) END_RECORDS = 'YES';
    END_RECORDS = 'NO ';
  /* READ PART */
  READ FILE(TAPEIN) INTO(T_R);
  DO WHILE (END_RECORDS = 'NO ');
    CALL SET_RTN;
    READ FILE(TAPEIN) INTO(T_R);
  END ;
  CALL SUM_RTN ;
  CALL WRITE_RTN;
  CLOSE FILE(TAPEIN),FILE(TAPEOUT);
  /* DECLARE PART */
DCL TAPEIN FILE RECORD INPUT;
DCL 1 T_R,
  3 F1 CHAR(24),
  3 MDO PIC '99',
  3 GUN PIC '99',
  3 FDO PIC '99',
  3 F2 CHAR(10),
  3 YR CHAR(2),
  3 WOL PIC '99',
  3 F3 CHAR(8),
  3 MHJ PIC '9',
  3 FHJ PIC '9',
  3 MAG PIC '99',
  3 FAG PIC '99',
  3 F4 CHAR(22);
DCL TAPEOUT FILE RECORD OUTPUT;
DCL 1 TA_R,
  3 PYO PIC '99',
  3 PYR PIC '99',
  3 PJUL PIC '999',
  3 PCAN PIC '99',
  3 TAA(40) PIC '(8)9';
DCL YEAR(10) CHAR(2) INIT('81','80','79','78','77',
  '76','75','74','73','72');
DCL AGE111(30) PIC '99' INIT(00,15,20,21,22,23,24,25,26,27,28,29,30,
  31,32,33,34,35,36,37,38,39,40,41,42,43,
  44,45,50,55);
DCL AGE112(30) PIC '99' INIT(00,15,16,17,18,19,20,21,22,23,24,25,26,
  27,28,29,30,31,32,33,34,35,36,37,38,39,
  40,45,50,55);

```

SVM00010  
SVM00020  
SVM00030  
SVM00040  
SVM00050  
SVM00060  
SVM00070  
SVM00080  
SVM00090  
SVM00100  
SVM00110  
SVM00120  
SVM00130  
SVM00140  
SVM00150  
SVM00160  
SVM00170  
SVM00180  
SVM00190  
SVM00200  
SVM00210  
SVM00220  
SVM00230  
SVM00240  
SVM00250  
SVM00260  
SVM00270  
SVM00280  
SVM00290  
SVM00300  
SVM00310  
SVM00320  
SVM00330  
SVM00340  
SVM00350  
SVM00360  
SVM00370  
SVM00380  
SVM00390  
SVM00400  
SVM00410  
SVM00420  
SVM00430  
SVM00440  
SVM00450  
SVM00460  
SVM00470  
SVM00480  
SVM00490  
SVM00500  
SVM00510  
SVM00520  
SVM00530  
SVM00540  
SVM00550

```

DCL AGE12(63) PIC '99' INIT(00,15,16,17,18,19,20,21,22,23,24,25,26, SVM00560
27,28,29,30,31,32,33,34,35,36,37,38,39, SVM00570
40,41,42,43,44,45,46,47,48,49,50,51,52, SVM00580
53,54,55,56,57,58,59,60,61,62,63,64,65, SVM00590
66,67,68,69,70,71,72,73,74,75,76); SVM00600
DCL AGES(9) PIC '99' INIT(0,15,20,25,30,35,40,45,50); SVM00610
DCL DOD(14) PIC '99' INIT(0,11,21,22,23,31,32,33,34,35,36,37,38,39); SVM00620
DCL TAB10(10,13,15) FIXED DEC(8); SVM00630
DCL TAB11(10,32,32) FIXED DEC(8); SVM00640
DCL TAB12(10,65,15,2) FIXED DEC(8); SVM00650
DCL TAB13(15,4,10,4) FIXED DEC(8); SVM00660
DCL END_RECORDS CHAR(3); SVM00670
/* SUB PROCEDURE */ SVM00680
SET_RTN: PROCEDURE; SVM00690
DO N=1 BY 1 UNTIL(YR = YEAR(N) | N > 10); END; SVM00700
IF N > 10 THEN GOTO SET6; SVM00710
I=N; SVM00720
/* WOL PROCEDURE */ SVM00730
DO N = 1 BY 1 UNTIL(WOL = N | N > 11); END; SVM00740
J10=N+1; SVM00750
SVM00760
DO N=2 BY 1 UNTIL(MDO = DOD(N) | N > 14); END; SVM00770
K10=N; SVM00780
SVM00790
/**/ DO N=2 BY 1 UNTIL(MAG < AGE11(N) | N > 30); END; SVM00800
J11=N; SVM00810
/**/ DO N = 2 BY 1 UNTIL(FAG < AGE12(N) | N > 30); END; SVM00820
K11=N; SVM00830
SVM00840
DO N = 2 BY 1 UNTIL(MAG < AGE12(N) | N > 63); END; SVM00850
J121 = N; SVM00860
DO N = 2 BY 1 UNTIL(FAG < AGE12(N) | N > 63); END; SVM00870
J122=N; SVM00880
K121=K10; SVM00890
DO N = 2 BY 1 UNTIL(FDO = DOD(N) | N > 14); END; SVM00900
K122=N; SVM00910
I13=K10; SVM00920
J13=MHJ+1; SVM00930
K13=I; SVM00940
L13=FHJ+1; SVM00950
TAB10(I, J10, K10) = TAB10(I, J10, K10)+1; SVM00960
TAB11(I, J11, K11) = TAB11(I, J11, K11)+1; SVM00970
TAB11(I, 1, K11) = TAB11(I, 1, K11)+1; SVM00980
TAB12(I, J121, K121, 1) = TAB12(I, J121, K121, 1)+1; SVM00990
TAB12(I, 1, K121, 1) = TAB12(I, 1, K121, 1)+1; SVM01000
TAB12(I, J122, K122, 2) = TAB12(I, J122, K122, 2)+1; SVM01010
TAB12(I, 1, K122, 2) = TAB12(I, 1, K122, 2)+1; SVM01020
TAB13(I13, J13, K13, L13) = TAB13(I13, J13, K13, L13)+1; SVM01030
TAB13(1, J13, K13, L13) = TAB13(1, J13, K13, L13)+1; SVM01040
END SET_RTN; SVM01050
SET6: SVM01060
SUM_RTN: PROC; SVM01070
DO I=2 TO 13; SVM01080
TAB10(*,1,*) = TAB10(*,1,*) + TAB10(*,I,*); SVM01090
END; SVM01100
DO I=2 TO 15;

```

```

        TAB10(*,*,1)=TAB10(*,*,1)+TAB10(*,*,I);
    END;
    DO I=2 TO 32;
        TAB11(*,*,1) = TAB11(*,*,1) + TAB11(*,*,I);
    END;
    DO I=2 TO 15;
        TAB12(*,*,1,1)=TAB12(*,*,1,1)+TAB12(*,*,I,1);
        TAB12(*,*,1,2)=TAB12(*,*,1,2)+TAB12(*,*,I,2);
    END;
    DO I =2 TO 4;
        TAB13(*,1,*,*)=TAB13(*,1,*,*)+ TAB13(*,I,*,*);
    END;
    DO I=2 TO 4;
        TAB13(*,*,*,1) = TAB13(*,*,*,1) + TAB13(*,*,*,I);
    END;
END SUM_RTN;
WRITE_RTN: PROCEDURE ;
    CALL SUBWR (TAB10,10,10,13,15);
    CALL SUBWR (TAB11,11,10,32,32);
    N=0;
    DO I=1 TO 10;
        DO J=1 TO 65;
            PYO=12;
            PYR=I;
            PJUL=J;
            DO K=1 TO 15;
                DO L=1 TO 2;
                    N=N+1;
                    TAA(N)=TAB12(I,J,K,L);
                END;
            END;
            WRITE FILE(TAPEOUT) FROM (TA_R);
            N=0;
        END;
    END;
    N=0;
    NN=0;
    DO I=1 TO 15;
        DO J=1 TO 4;
            NN=NN+1;
            PYO=13;
            PYR=NN;
            PJUL=J;
            DO K=1 TO 10;
                DO L=1 TO 4;
                    N=N+1;
                    TAA(N)=TAB13(I,J,K,L);
                END;
            END;
            WRITE FILE(TAPEOUT) FROM (TA_R);
            N=0;
        END;
    END;
END WRITE_RTN;
SUBWR: PROC (ARRAY,PP,II,JJ,KK);

```

SVM01110  
 SVM01120  
 SVM01130  
 SVM01140  
 SVM01150  
 SVM01160  
 SVM01170  
 SVM01180  
 SVM01190  
 SVM01200  
 SVM01210  
 SVM01220  
 SVM01230  
 SVM01240  
 SVM01250  
 SVM01260  
 SVM01270  
 SVM01280  
 SVM01290  
 SVM01300  
 SVM01310  
 SVM01320  
 SVM01330  
 SVM01340  
 SVM01350  
 SVM01360  
 SVM01370  
 SVM01380  
 SVM01390  
 SVM01400  
 SVM01410  
 SVM01420  
 SVM01430  
 SVM01440  
 SVM01450  
 SVM01460  
 SVM01470  
 SVM01480  
 SVM01490  
 SVM01500  
 SVM01510  
 SVM01520  
 SVM01530  
 SVM01540  
 SVM01550  
 SVM01560  
 SVM01570  
 SVM01580  
 SVM01590  
 SVM01600  
 SVM01610  
 SVM01620  
 SVM01630  
 SVM01640  
 SVM01650



DCL ARRAY(*,*,*) FIXED DEC(8);	SVM01660
DO I= 1 TO II;	SVM01670
DO J=1 TO JJ;	SVM01680
PYO=PP;	SVM01690
PYR=I;	SVM01700
PJUL=J;	SVM01710
DO K=1 TO KK;	SVM01720
TAA(K) = ARRAY(I,J,K);	SVM01730
END;	SVM01740
WRITE FILE(TAPEOUT) FROM(TA_R);	SVM01750
END;	SVM01760
END;	SVM01770
END SUBWR;	SVM01780
END CHOI;	SVM01790
/*	SVM01800
//GO.TAPEIN DD UNIT=TAPE,DISP=(OLD,KEEP),VOL=SER=CHOI13,	SVM01810
//          DSN=CHOI13,LABEL=(2,BLF),	SVM01820
//          DCB=(LRECL=80,RECFM=FB,BLKSIZE=3200,EROPT=SKP)	SVM01830
//GO.TAPEOUT DD UNIT=TAPE,DISP=(NEW,KEEP),DSN=CHOI1,	SVM01840
//          DCB=(LRECL=329,RECFM=FB,BLKSIZE=3290,EROPT=SKP)	SVM01850
//GO.SYSPRINT DD SYSOUT=A	SVM01860
//GO.SYSOUT DD SYSOUT=A	SVM01870
//	SVM01880

```

//SVMWDL14 JOB CLASS=S, TYPRUN=HOLD
/** MARRIAGE TABLE SUMMARY TO TAPE SUM ADD UP *****
// EXEC PLIFCLG
//PLI.SYSIN DD *
* PROCESS GS, NEST, OPT(TIME);
  CHOI: PROC OPTIONS(MAIN);
    TAB10=0;
    TAB11=0;
    TAB12=0;
    TAB13=0;
    OPEN FILE(TAPEIN1) INPUT,
          FILE(TAPEIN2) INPUT,
          FILE(TAPEOUT) OUTPUT,
          FILE(SYSPRINT) LINESIZE(132) PAGESIZE(60);
    ON ENDFILE(TAPEIN1) END1_REC='YES';
    ON ENDFILE(TAPEIN2) END2_REC='YES';
    END1_REC = 'NO';
    END2_REC = 'NO';
/* TAPE READ PART */
  READ FILE(TAPEIN1) INTO(T_R);
  DO WHILE(END1_REC = 'NO');
    IF GU = 10 THEN CALL SET10_R;
    ELSE
      IF GU = 11 THEN CALL SET11_R;
      ELSE
        IF GU = 12 THEN CALL SET12_R;
        ELSE CALL SET13_R;
        READ FILE(TAPEIN1) INTO(T_R);
      END;
    J=0;
    READ FILE(TAPEIN2) INTO(TI_R);
    DO WHILE(END2_REC = 'NO');
      IF KU = 10 THEN CALL SET10_RTN;
      ELSE
        IF KU = 11 THEN CALL SET11_RTN;
        ELSE
          IF KU = 12 THEN CALL SET12_RTN;
          ELSE CALL SET13_RTN;
          READ FILE(TAPEIN2) INTO(TI_R);
        END;
      CALL WRITE_RTN;
    CLOSE FILE(TAPEIN1), FILE(TAPEIN2), FILE(TAPEOUT);
/* DECLARE PART */
  DCL TAPEIN1 FILE RECORD INPUT;
  DCL 1 T_R,
      3 GU PIC '99',
      3 GU1 PIC '99',
      3 GU2 PIC '999',
      3 GU3 PIC '99',
      3 AA(40) PIC '(8)9';
  DCL TAPEIN2 FILE RECORD INPUT;
  DCL 1 TI_R,
      3 KU PIC '99',
      3 KU1 PIC '99',
      3 KU2 PIC '999',
      3 KU3 PIC '99',

```

SVM00010  
 SVM00020  
 SVM00030  
 SVM00040  
 SVM00050  
 SVM00060  
 SVM00070  
 SVM00080  
 SVM00090  
 SVM00100  
 SVM00110  
 SVM00120  
 SVM00130  
 SVM00140  
 SVM00150  
 SVM00160  
 SVM00170  
 SVM00180  
 SVM00190  
 SVM00200  
 SVM00210  
 SVM00220  
 SVM00230  
 SVM00240  
 SVM00250  
 SVM00260  
 SVM00270  
 SVM00280  
 SVM00290  
 SVM00300  
 SVM00310  
 SVM00320  
 SVM00330  
 SVM00340  
 SVM00350  
 SVM00360  
 SVM00370  
 SVM00380  
 SVM00390  
 SVM00400  
 SVM00410  
 SVM00420  
 SVM00430  
 SVM00440  
 SVM00450  
 SVM00460  
 SVM00470  
 SVM00480  
 SVM00490  
 SVM00500  
 SVM00510  
 SVM00520  
 SVM00530  
 SVM00540  
 SVM00550

```

        3 BB(40) PIC '(8)9';
DCL TAPEOUT FILE RECORD OUTPUT;
DCL 1 TA_R,
    3 GUA PIC '99',
    3 GUB PIC '99',
    3 GUC PIC '999',
    3 GUD PIC '99',
    3 CC(40) PIC '(8)9';
DCL TAB10(10,13,15) FIXED DEC(8);
DCL TAB11(10,32,32) FIXED DEC(8);
DCL TAB12(10,65,30) FIXED DEC(8);
DCL TAB13(60,40) FIXED DEC(8);
DCL END1_REC CHAR(3);
DCL END2_REC CHAR(3);
/**** SUB PROCEDURE ****/
SET10_R: PROC;
    IF GU1 = 10 THEN GOTO SET10_E;
    I=GU1+1;
    J=GU2;
    DO K=1 TO 15;
        TAB10(I,J,K) = TAB10(I,J,K) + AA(K);
    END;
SET10_E: END SET10_R;
SET11_R: PROC;
    IF GU1=10 THEN GOTO SET11_E;
    I=GU1+1;
    J=GU2;
    DO K=1 TO 32;
        TAB11(I,J,K) = TAB11(I,J,K) + AA(K);
    END;
SET11_E: END SET11_R;
SET12_R: PROC;
    IF GU1 = 10 THEN GOTO SET12_E;
    I=GU1+1;
    J=GU2;
    DO K=1 TO 30;
        TAB12(I,J,K) = TAB12(I,J,K) + AA(K);
    END;
SET12_E: END SET12_R;
SET13_R: PROC;
    I=GU1;
    DO J = 1 TO 36;
        N=J+4;
        TAB13(I,N) = TAB13(I,N) + AA(J);
    END;
SET13_E: END SET13_R;
SET10_RTN: PROC;
    I=KU1;
    J=KU2;
    DO K=1 TO 15;
        TAB10(I,J,K)=TAB10(I,J,K) + BB(K);
    END;
END SET10_RTN;
SET11_RTN: PROC;
    I=KU1;

```

SVM00560  
SVM00570  
SVM00580  
SVM00590  
SVM00600  
SVM00610  
SVM00620  
SVM00630  
SVM00640  
SVM00650  
SVM00660  
SVM00670  
SVM00680  
SVM00690  
SVM00700  
SVM00710  
SVM00720  
SVM00730  
SVM00740  
SVM00750  
SVM00760  
SVM00770  
SVM00780  
SVM00790  
SVM00800  
SVM00810  
SVM00820  
SVM00830  
SVM00840  
SVM00850  
SVM00860  
SVM00870  
SVM00880  
SVM00890  
SVM00900  
SVM00910  
SVM00920  
SVM00930  
SVM00940  
SVM00950  
SVM00960  
SVM00970  
SVM00980  
SVM00990  
SVM01000  
SVM01010  
SVM01020  
SVM01030  
SVM01040  
SVM01050  
SVM01060  
SVM01070  
SVM01080  
SVM01090  
SVM01100

```

        J=KU2;
        DO K=1 TO 32;
            TAB11(I,J,K)=TAB11(I,J,K) + BB(K);
        END;
    END SET11_RTN;
SET12_RTN:  PROC;
        I=KU1;
        J=KU2;
        DO K=1 TO 30;
            TAB12(I,J,K)= TAB12(I,J,K) + BB(K);
        END;
    END SET12_RTN;
SET13_RTN:  PROC;
        I=KU1;
        DO J=1 TO 40;
            TAB13(I,J)=TAB13(I,J) + BB(J);
        END;
    END SET13_RTN;
WRITE_RTN:  PROC;
        CALL SUBWR(TAB10,10,13,15,10) ;
        CALL SUBWR(TAB11,10,32,32,11) ;
        CALL SUBWR(TAB12,10,65,30,12) ;
        DO I=1 TO 60;
            DO J=1 TO 40;
                GUA=13;
                GUB=I;
                GUC=J;
                CC(J) = TAB13(I,J);
            END;
            WRITE FILE(TAPEOUT) FROM(TA_R);
        END;
    END WRITE_RTN;
SUBWR:  PROC (ARRAY,II,JJ,KK,PP);
DCL    ARRAY(*,*,*) FIXED DEC(8);
        DO I=1 TO II;
            DO J=1 TO JJ;
                GUA=PP;
                GUB=I;
                GUC=J;
                DO K=1 TO KK;
                    CC(K) = ARRAY(I,J,K);
                END;
                WRITE FILE(TAPEOUT) FROM(TA_R);
            END;
        END;
    END SUBWR;
END CHOI;
/*
//GO.TAPEIN1 DD DSN=SVMWED,UNIT=TAPE,DISP=(OLD,KEEP),VOL=SER=UU1619,
//            DCB=(RECFM=FB,LRECL=329,BLKSIZE=3290,EROPT=SKP),
//            LABEL=(2,BLP)
//GO.TAPEIN2 DD DSN=SVMWET,UNIT=TAPE,DISP=(OLD,KEEP),VOL=SER=AU3750,
//            DCB=(RECFM=FB,LRECL=329,BLKSIZE=3290,EROPT=SKP),
//            LABEL=(2,BLP)
//GO.TAPEOUT DD DSN=SVMWMT,UNIT=TAPE,DISP=(NEW,KEEP),

```

SVM01110  
 SVM01120  
 SVM01130  
 SVM01140  
 SVM01150  
 SVM01160  
 SVM01170  
 SVM01180  
 SVM01190  
 SVM01200  
 SVM01210  
 SVM01220  
 SVM01230  
 SVM01240  
 SVM01250  
 SVM01260  
 SVM01270  
 SVM01280  
 SVM01290  
 SVM01300  
 SVM01310  
 SVM01320  
 SVM01330  
 SVM01340  
 SVM01350  
 SVM01360  
 SVM01370  
 SVM01380  
 SVM01390  
 SVM01400  
 SVM01410  
 SVM01420  
 SVM01430  
 SVM01440  
 SVM01450  
 SVM01460  
 SVM01470  
 SVM01480  
 SVM01490  
 SVM01500  
 SVM01510  
 SVM01520  
 SVM01530  
 SVM01540  
 SVM01550  
 SVM01560  
 SVM01570  
 SVM01580  
 SVM01590  
 SVM01600  
 SVM01610  
 SVM01620  
 SVM01630  
 SVM01640  
 SVM01650

FILE: SVMWDL14 ADDUPM A1 VM/SP RELEASE 2.1 EXPRESS PUT8302+ SLU203

```
//          DCB=(RECFM=FB,LRECL=329,BLKSIZE=3290,EROPT=SKP)          SVM01660
//GO.SYSPRINT DD SYSOUT=A          SVM01670
//GO.SYSOUT DD SYSOUT=A          SVM01680
//          SVM01690
```

```

//SVMWTE14 JOB TYPRUN=HOLD
//TABED EXEC COBUCLG,PARM='NOSOURCE'
//COB.SYSIN DD *
  ID DIVISION.
  PROGRAM-ID. MARED.
  ENVIRONMENT DIVISION.
  CONFIGURATION SECTION.
  SPECIAL-NAMES.
    C01 IS CHOI
    CSP IS DOUB.
  INPUT-OUTPUT SECTION.
  FILE-CONTROL.
    SELECT T-F ASSIGN UT-S-TAPEIN.
    SELECT P-F ASSIGN UT-S-PROUT.
  DATA DIVISION.
  FILE SECTION.
  FD T-F BLOCK CONTAINS 10 RECORDS
    LABEL RECORD STANDARD.
01 T-R.
  03 GU PIC 99.
  03 GU1 PIC 99.
  03 GU2 PIC 999.
  03 GU3 PIC 99.
  03 AA OCCURS 40 PIC 9(8).
  FD P-F LABEL RECORD OMITTED
01 P-R PIC X(132).
WORKING-STORAGE SECTION.
  77 I PIC 9(3) VALUE 0.
  77 J PIC 9(3) VALUE 0.
  77 K PIC 9(3) VALUE 0.
  77 L PIC 9(3) VALUE 0.
  77 M PIC 9(3) VALUE 1.
  77 N PIC 9(3) VALUE 2.
  77 B1 PIC 9(3) VALUE 1.
  77 B2 PIC 9(3) VALUE 2.
  77 B3 PIC 9(3) VALUE 5.
  77 D1 PIC 99 VALUE 0.
  77 B4 PIC 9(3) VALUE 0.
  77 S11 PIC 9 VALUE 0.
  77 S12 PIC 9 VALUE 0.
  77 S13 PIC 9 VALUE 0.
  77 CNT PIC 9(3) VALUE 0.
01 LIST10.
  03 F PIC X(15) VALUE SPACE.
  03 P10 OCCURS 13 PIC Z(7)9.
  03 P101 PIC Z(6)9.
  03 P102 PIC Z(5)9.
01 LIST111.
  03 F PIC X(34) VALUE SPACE.
  03 P111 OCCURS 7 PIC Z(13)9.
01 LIST112.
  03 F PIC X(4) VALUE SPACE.
  03 P112 OCCURS 9 PIC Z(13)9.
01 LIST121.
  03 F PIC X(18) VALUE SPACE.

```

SVM00010  
 SVM00020  
 SVM00030  
 SVM00040  
 SVM00050  
 SVM00060  
 SVM00070  
 SVM00080  
 SVM00090  
 SVM00100  
 SVM00110  
 SVM00120  
 SVM00130  
 SVM00140  
 SVM00150  
 SVM00160  
 SVM00170  
 SVM00180  
 SVM00190  
 SVM00200  
 SVM00210  
 SVM00220  
 SVM00230  
 SVM00240  
 SVM00250  
 SVM00260  
 SVM00270  
 SVM00280  
 SVM00290  
 SVM00300  
 SVM00310  
 SVM00320  
 SVM00330  
 SVM00340  
 SVM00350  
 SVM00360  
 SVM00370  
 SVM00380  
 SVM00390  
 SVM00400  
 SVM00410  
 SVM00420  
 SVM00430  
 SVM00440  
 SVM00450  
 SVM00460  
 SVM00470  
 SVM00480  
 SVM00490  
 SVM00500  
 SVM00510  
 SVM00520  
 SVM00530  
 SVM00540  
 SVM00550

	03 PTM	PIC Z(8)9.	SVM00560
	03 PTF	PIC Z(8)9.	SVM00570
	03 P121	OCCURS 12 PIC Z(7)9.	SVM00580
01	LIST122.		SVM00590
	03 P122	OCCURS 16 PIC Z(7)9.	SVM00600
01	LIST131.		SVM00610
	03 F	PIC X(44) VALUE SPACE.	SVM00620
	03 P131	OCCURS 8 PIC Z(10)9.	SVM00630
01	LIST132.		SVM00640
	03 P132	OCCURS 12 PIC Z(9)9.	SVM00650
01	TABL11.		SVM00660
	03 TAB11	OCCURS 32.	SVM00670
	05 TA11	OCCURS 32 PIC 9(8).	SVM00680
01	TABL12.		SVM00690
	03 TAB12	OCCURS 65.	SVM00700
	05 TA12	OCCURS 30 PIC 9(8).	SVM00710
01	TABL13.		SVM00720
	03 TAB13	OCCURS 60.	SVM00730
	05 TA13	OCCURS 40 PIC 9(8).	SVM00740
01	BIGYO	PIC X(6) VALUE "020610".	SVM00750
01	BGYO	REDEFINES BIGYO.	SVM00760
	03 B13	OCCURS 3 PIC 99.	SVM00770
	PROCEDURE	DIVISION.	SVM00780
	OPEN INPUT	T-F.	SVM00790
	OPEN OUTPUT	P-F.	SVM00800
	BUSAN.		SVM00810
	MOVE SPACE	TO P-R.	SVM00820
	WRITE P-R	AFTER ADVANCING CHOI.	SVM00830
	MOVE SPACE	TO P-R.	SVM00840
	WRITE P-R	AFTER ADVANCING 3.	SVM00850
	MOVE SPACE	TO P-R.	SVM00860
	WRITE P-R	AFTER ADVANCING 3.	SVM00870
	MOVE SPACE	TO P-R.	SVM00880
	WRITE P-R	AFTER ADVANCING 3.	SVM00890
	MOVE SPACE	TO P-R.	SVM00900
	WRITE P-R	AFTER ADVANCING 1.	SVM00910
	MOVE SPACE	TO P-R.	SVM00920
	BUSAN1.		SVM00930
	WRITE P-R	AFTER ADVANCING 3.	SVM00940
	MOVE SPACE	TO P-R.	SVM00950
	WRITE P-R	AFTER ADVANCING 1.	SVM00960
	MOVE SPACE	TO P-R.	SVM00970
	READ-RTN.		SVM00980
	READ T-F	AT END GO TO P13-RTN.	SVM00990
	COMPUTE D1	= GU - 9.	SVM01000
	GO TO 10T 11T 12T 13T	DEPENDING ON D1.	SVM01010
	10T.		SVM01020
	ADD 1	TO CNT.	SVM01030
	IF CNT = 66	PERFORM BUSAN THRU BUSAN1 MOVE 1 TO CNT M.	SVM01040
	PERFORM 10A	VARYING I FROM 1 BY 1 UNTIL I > 13.	SVM01050
	MOVE AA(14)	TO P101.	SVM01060
	MOVE AA(15)	TO P102.	SVM01070
	IF CNT = M	WRITE P-R FROM LIST10 AFTER ADVANCING 3	SVM01080
		WRITE P-R FROM LIST10 AFTER ADVANCING DOUB	SVM01090
		MOVE SPACE TO P-R	SVM01100

COMPUTE M = M + 13	SVM01110
GO TO READ-RTN.	SVM01120
PERFORM A1 VARYING I FROM 1 BY 1 UNTIL I > 3.	SVM01130
WRITE P-R FROM LIST10 AFTER ADVANCING 1.	SVM01140
MOVE SPACE TO P-R.	SVM01150
GO TO READ-RTN.	SVM01160
A1. IF GU2 = B13 (I). GO TO L2-RTN.	SVM01170
L2-RTN.	SVM01180
WRITE P-R FROM LIST10 AFTER ADVANCING 2.	SVM01190
MOVE SPACE TO P-R.	SVM01200
GO TO READ-RTN.	SVM01210
10A. MOVE AA(I) TO P10 (I).	SVM01220
11T.	SVM01230
11. IF GU1 NOT = B1 GO TO 11T-PRT.	SVM01240
RT. MOVE GU2 TO I.	SVM01250
PERFORM A11 VARYING J FROM 1 BY 1 UNTIL J > 32.	SVM01260
GO TO READ-RTN.	SVM01270
A11. MOVE AA (J) TO TA11 (I J).	SVM01280
11T-PRT.	SVM01290
PERFORM BUSAN.	SVM01300
PERFORM 11A THRU EX VARYING I FROM 1 BY 1 UNTIL I > 32.	SVM01310
MOVE 2 TO B2.	SVM01320
PERFORM BUSAN.	SVM01330
PERFORM 11C THRU ET VARYING I FROM 1 BY 1 UNTIL I > 32.	SVM01340
22. MOVE 2 TO B2.	SVM01350
WRITE P-R AFTER ADVANCING CHOI. MOVE SPACE TO P-R	SVM01360
33. WRITE P-R AFTER ADVANCING 2.	SVM01370
MOVE SPACE TO P-R.	SVM01380
PP. PERFORM 11E THRU EJ VARYING I FROM 1 BY 1 UNTIL I > 32.	SVM01390
PERFORM 22 THRU 33.	SVM01400
PERFORM 11H THRU EK VARYING I FROM 1 BY 1 UNTIL I > 32.	SVM01410
MOVE 2 TO B2. MOVE GU1 TO B1.	SVM01420
PERFORM BUSAN.	SVM01430
PA. GO TO RT.	SVM01440
11A.	SVM01450
PERFORM 11B VARYING J FROM 1 BY 1 UNTIL J > 7.	SVM01460
PC. IF I = 1 WRITE P-R FROM LIST111 AFTER ADVANCING 2	SVM01470
WRITE P-R FROM LIST111 AFTER ADVANCING DOUB	SVM01480
MOVE SPACE TO P-R	SVM01490
GO TO EX.	SVM01500
IF I = B2 WRITE P-R FROM LIST111 AFTER ADVANCING 2	SVM01510
MOVE SPACE TO P-R	SVM01520
COMPUTE B2 = B2 + 3	SVM01530
GO TO EX.	SVM01540
WRITE P-R FROM LIST111 AFTER ADVANCING 1.	SVM01550
MOVE SPACE TO P-R.	SVM01560
EX. EXIT.	SVM01570
11B. MOVE TA11 (I J) TO P111 (J).	SVM01580
11C. PERFORM 11D VARYING J FROM 8 BY 1 UNTIL J > 16.	SVM01590
PD. IF I = 1 WRITE P-R FROM LIST112 AFTER ADVANCING 2	SVM01600
WRITE P-R FROM LIST112 AFTER ADVANCING DOUB	SVM01610
MOVE SPACE TO P-R	SVM01620
GO TO ET.	SVM01630
IF I = B2 WRITE P-R FROM LIST112 AFTER ADVANCING 2	SVM01640
MOVE SPACE TO P-R	SVM01650



COMPUTE B2 = B2 + 3	SVM01660
GO TO ET.	SVM01670
WRITE P-R FROM LIST112 AFTER ADVANCING 1.	SVM01680
MOVE SPACE TO P-R.	SVM01690
ET. EXIT.	SVM01700
11D. COMPUTE K = J - 7.	SVM01710
MOVE TA11 (I J) TO P112 (K).	SVM01720
11E. PERFORM 11F VARYING J FROM 17 BY 1 UNTIL J > 23.	SVM01730
IF I = 1 WRITE P-R FROM LIST111 AFTER ADVANCING 2	SVM01740
WRITE P-R FROM LIST111 AFTER ADVANCING DOUB	SVM01750
MOVE SPACE TO P-R	SVM01760
GO TO EJ.	SVM01770
IF I = B2 WRITE P-R FROM LIST111 AFTER ADVANCING 2	SVM01780
MOVE SPACE TO P-R	SVM01790
COMPUTE B2 = B2 + 3	SVM01800
GO TO EJ.	SVM01810
WRITE P-R FROM LIST111 AFTER ADVANCING 1.	SVM01820
MOVE SPACE TO P-R.	SVM01830
EJ. EXIT.	SVM01840
11F. COMPUTE K = J - 16.	SVM01850
MOVE TA11 (I J) TO P111 (K).	SVM01860
11H. PERFORM 11J VARYING J FROM 24 BY 1 UNTIL J > 32.	SVM01870
IF I = 1 WRITE P-R FROM LIST112 AFTER ADVANCING 2	SVM01880
WRITE P-R FROM LIST112 AFTER ADVANCING DOUB	SVM01890
MOVE SPACE TO P-R	SVM01900
GO TO EK.	SVM01910
IF I = B2 WRITE P-R FROM LIST112 AFTER ADVANCING 2	SVM01920
MOVE SPACE TO P-R	SVM01930
COMPUTE B2 = B2 + 3	SVM01940
GO TO EK.	SVM01950
WRITE P-R FROM LIST112 AFTER ADVANCING 1.	SVM01960
MOVE SPACE TO P-R.	SVM01970
EK. EXIT.	SVM01980
11J. COMPUTE K = J - 23.	SVM01990
MOVE TA11 (I J) TO P112 (K).	SVM02000
12T.	SVM02010
IF S12 = 1 GO TO 12T-BG.	SVM02020
PERFORM 11T-PRT THRU PP.	SVM02030
MOVE 1 TO S12.	SVM02040
MOVE 1 TO B1.	SVM02050
12T-BG.	SVM02060
IF GU1 NOT = B1 GO TO 12T-PRT.	SVM02070
12T-RT.	SVM02080
MOVE GU2 TO I.	SVM02090
PERFORM A12 VARYING J FROM 1 BY 1 UNTIL J > 30.	SVM02100
GO TO READ-RTN.	SVM02110
A12. MOVE AA (J) TO TA12 (I J).	SVM02120
12T-PRT.	SVM02130
PERFORM 12A THRU EE VARYING I FROM 1 BY 1 UNTIL I > 65.	SVM02140
MOVE 5 TO B3.	SVM02150
PERFORM BUSAN THRU BUSAN1.	SVM02160
PERFORM 12C THRU EP VARYING I FROM 1 BY 1 UNTIL I > 65.	SVM02170
PERFORM BUSAN THRU BUSAN1.	SVM02180
PB. MOVE 5 TO B3.	SVM02190
MOVE GU1 TO B1.	SVM02200

GO TO 12T-RT.	SVM02210
12A. MOVE TA12 (I 1) TO PTM.	SVM02220
MOVE TA12 (I 2) TO PTF.	SVM02230
PERFORM 12B	SVM02240
IF I = 1	VARYING J FROM 3 BY 1 UNTIL J > 14.
WRITE P-R FROM LIST121	AFTER ADVANCING 3
WRITE P-R FROM LIST121	AFTER ADVANCING DOUB
MOVE SPACE TO P-R	
GO TO EE.	
IF I = 2	WRITE P-R FROM LIST121
WRITE P-R FROM LIST121	AFTER ADVANCING 3
MOVE SPACE TO P-R	
GO TO EE.	
IF I = B3	WRITE P-R FROM LIST121
WRITE P-R FROM LIST121	AFTER ADVANCING 2
MOVE SPACE TO P-R	
COMPUTE B3 = B3 + 3	
GO TO EE.	
WRITE P-R FROM LIST121	AFTER ADVANCING 1
MOVE SPACE TO P-R.	
EE. EXIT.	
12B. COMPUTE K = J - 2	
MOVE TA12 (I J) TO P121 (K).	
12C. PERFORM 12D	VARYING J FROM 15 BY 1 UNTIL J > 30.
IF I = 1	WRITE P-R FROM LIST122
WRITE P-R FROM LIST122	AFTER ADVANCING 3
MOVE SPACE TO P-R	
GO TO EP.	
IF I = 2	WRITE P-R FROM LIST122
WRITE P-R FROM LIST122	AFTER ADVANCING 3
MOVE SPACE TO P-R	
GO TO EP.	
IF I = B3	WRITE P-R FROM LIST122
WRITE P-R FROM LIST122	AFTER ADVANCING 2
MOVE SPACE TO P-R	
COMPUTE B3 = B3 + 3	
GO TO EP.	
WRITE P-R FROM LIST122	AFTER ADVANCING 1.
MOVE SPACE TO P-R.	
EP. EXIT.	
12D. COMPUTE K = J - 14.	
MOVE TA12 (I J) TO P122 (K).	
13T. IF S13 = 1	GO TO 13T-BG.
PERFORM 12T-PRT.	
MOVE 1 TO S13 B1.	
MOVE 2 TO B2.	
13T-BG. MOVE GU1 TO I.	
PERFORM A13	VARYING J FROM 1 BY 1 UNTIL J > 40.
GO TO READ-RTN.	
A13. MOVE AA (J) TO TA13 (I J).	
P13-RTN. MOVE 1 TO B1.	
MOVE 2 TO B2.	
PERFORM BUSAN.	
PERFORM PT13 THRU X1	VARYING I FROM 1 BY 1 UNTIL I > 60.
MOVE 1 TO B1.	

MOVE 2 TO B2.	SVM02760
PERFORM BUSAN.	SVM02770
PERFORM PTH THRU X2 VARYING I FROM 1 BY 1 UNTIL I > 60.	SVM02780
MOVE 1 TO B1.	SVM02790
MOVE 2 TO B2.	SVM02800
PERFORM BUSAN.	SVM02810
PERFORM PTRR THRU X3 VARYING I FROM 1 BY 1 UNTIL I > 60.	SVM02820
MOVE 1 TO B1.	SVM02830
MOVE 2 TO B2.	SVM02840
PERFORM BUSAN.	SVM02850
PERFORM PTTN THRU X4 VARYING I FROM 1 BY 1 UNTIL I > 60.	SVM02860
CLOSE T-F P-F.	SVM02870
STOP RUN.	SVM02880
PT13.	SVM02890
PERFORM P13 VARYING J FROM 1 BY 1 UNTIL J > 8.	SVM02900
IF I = B1 WRITE P-R FROM LIST131 AFTER ADVANCING 2	SVM02910
WRITE P-R FROM LIST131 AFTER ADVANCING DOUB	SVM02920
MOVE SPACE TO P-R	SVM02930
COMPUTE B1 = B1 + 4	SVM02940
GO TO X1.	SVM02950
IF I = B2 WRITE P-R FROM LIST131 AFTER ADVANCING 2	SVM02960
MOVE SPACE TO P-R	SVM02970
COMPUTE B2 = B2 + 4	SVM02980
GO TO X1.	SVM02990
WRITE P-R FROM LIST131 AFTER ADVANCING 1.	SVM03000
MOVE SPACE TO P-R.	SVM03010
X1. EXIT.	SVM03020
P13. MOVE TA13 (I J) TO P131 (J).	SVM03030
PTTH.	SVM03040
PERFORM PTH VARYING J FROM 9 BY 1 UNTIL J > 20.	SVM03050
IF I = B1 WRITE P-R FROM LIST132 AFTER ADVANCING 2	SVM03060
WRITE P-R FROM LIST132 AFTER ADVANCING DOUB	SVM03070
MOVE SPACE TO P-R	SVM03080
COMPUTE B1 = B1 + 4	SVM03090
GO TO X2.	SVM03100
IF I = B2 WRITE P-R FROM LIST132 AFTER ADVANCING 2	SVM03110
MOVE SPACE TO P-R	SVM03120
COMPUTE B2 = B2 + 4	SVM03130
GO TO X2.	SVM03140
WRITE P-R FROM LIST132 AFTER ADVANCING 1.	SVM03150
MOVE SPACE TO P-R.	SVM03160
X2. EXIT.	SVM03170
PTH. COMPUTE K = J - 8.	SVM03180
MOVE TA13 (I J) TO P132 (K).	SVM03190
PTTR.	SVM03200
PERFORM PTR VARYING J FROM 21 BY 1 UNTIL J > 28.	SVM03210
IF I = B1 WRITE P-R FROM LIST131 AFTER ADVANCING 2	SVM03220
WRITE P-R FROM LIST131 AFTER ADVANCING DOUB	SVM03230
MOVE SPACE TO P-R	SVM03240
COMPUTE B1 = B1 + 4	SVM03250
GO TO X3.	SVM03260
IF I = B2 WRITE P-R FROM LIST131 AFTER ADVANCING 2	SVM03270
MOVE SPACE TO P-R	SVM03280
COMPUTE B2 = B2 + 4	SVM03290
GO TO X3.	SVM03300

```

WRITE P-R FROM LIST131 AFTER ADVANCING 1. SVM03310
MOVE SPACE TO P-R. SVM03320
X3. EXIT. SVM03330
PTR. COMPUTE K = J - 20. SVM03340
MOVE TA13 (I J) TO P131 (K). SVM03350
PTTN. SVM03360
PERFORM PTTN VARYING J FROM 29 BY 1 UNTIL J > 40. SVM03370
IF I = B1 WRITE P-R FROM LIST132 AFTER ADVANCING 2 SVM03380
WRITE P-R FROM LIST132 AFTER ADVANCING DOUB SVM03390
MOVE SPACE TO P-R SVM03400
COMPUTE B1 = B1 + 4 SVM03410
GO TO X4. SVM03420
IF I = B2 WRITE P-R FROM LIST132 AFTER ADVANCING 2 SVM03430
MOVE SPACE TO P-R SVM03440
COMPUTE B2 = B2 + 4 SVM03450
GO TO X4. SVM03460
WRITE P-R FROM LIST132 AFTER ADVANCING 1. SVM03470
MOVE SPACE TO P-R. SVM03480
X4. EXIT. SVM03490
PTN. COMPUTE K = J - 28. SVM03500
MOVE TA13 (I J) TO P132 (K). SVM03510
SVM03520
SVM03530
/* SVM03540
//GO.TAPEIN DD UNIT=TAPE,DISP=OLD,DSN=VMSM59,VOL=SER=VM1111, SVM03550
// LABEL=(2,BLP),DCB=(EROPT=SKP) SVM03560
//SYSOUT DD SYSOUT=A SVM03570
//GO.PROUT DD SYSOUT=A SVM03580
//SYSDBOUT DD SYSOUT=A SVM03590
//

```

```

//SVIWT14 JOB TYPRUN=HOLD
//TTT EXEC COBUCLG
//COB.SYSIN DD *
ID DIVISION.
PROGRAM-ID. TTT.
ENVIRONMENT DIVISION.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT T-F ASSIGN UT-S-TAPEIN.
    SELECT TA-F ASSIGN UT-S-TAPEOUT.
DATA DIVISION.
FILE SECTION.
FD T-F LABEL RECORD OMITTED.
01 T-R.
    03 AA PIC X(13).
    03 BB PIC X(32).
    03 CC PIC X(3).
    03 DD REDEFINES CC PIC 999.
    03 F PIC X(32).
FD TA-F BLOCK CONTAINS 40 RECORDS
    LABEL RECORD STANDARD.
01 TA-R.
    02 TAA.
    03 TA PIC X(13).
    03 TC PIC 999.
    02 TBB.
    03 TB PIC X(32).
    03 F PIC X(32).
WORKING-STORAGE SECTION.
77 ECNT PIC 9(7) VALUE 0.
77 OCNT PIC 9(7) VALUE 0.
PROCEDURE DIVISION.
OPEN INPUT T-F.
OPEN OUTPUT TA-F.
MOVE SPACE TO TA-R.
READ-RTN.
READ T-F AT END GO TO END-RTN.
ADD 1 TO ECNT.
EXAMINE CC TALLYING ALL SPACE REPLACING BY 0.
IF DD NOT NUMERIC GO TO MOVE-RTN.
COMPUTE TC = DD / 10 ** TALLY.
MOVE-RTN.
MOVE AA TO TA.
MOVE BB TO TB.
WRITE TA-R.
MOVE SPACE TO TA-R.
ADD 1 TO OCNT.
GO TO READ-RTN.
END-RTN.
DISPLAY " ECNT = " ECNT.
DISPLAY " OCNT = " OCNT.
CLOSE T-F TA-F.
OPEN INPUT TA-F.
RD-RTN.
READ TA-F AT END CLOSE TA-F STOP RUN.

```

SVI00010  
SVI00020  
SVI00030  
SVI00040  
SVI00050  
SVI00060  
SVI00070  
SVI00080  
SVI00090  
SVI00100  
SVI00110  
SVI00120  
SVI00130  
SVI00140  
SVI00150  
SVI00160  
SVI00170  
SVI00180  
SVI00190  
SVI00200  
SVI00210  
SVI00220  
SVI00230  
SVI00240  
SVI00250  
SVI00260  
SVI00270  
SVI00280  
SVI00290  
SVI00300  
SVI00310  
SVI00320  
SVI00330  
SVI00340  
SVI00350  
SVI00360  
SVI00370  
SVI00380  
SVI00390  
SVI00400  
SVI00410  
SVI00420  
SVI00430  
SVI00440  
SVI00450  
SVI00460  
SVI00470  
SVI00480  
SVI00490  
SVI00500  
SVI00510  
SVI00520  
SVI00530  
SVI00540  
SVI00550

FILE: SVIWT114 RJTT A1 VM/SP RELEASE 2.1 EXPRESS PUT8302+ SLU203

```

        DISPLAY " " TAA.
        EX. PERFORM RD-RTN 1000 TIMES.
        CLOSE TA-F.
        STOP RUN.
/*
//GO.TAPEIN DD DSN=VBCHOI,UNIT=TAPE,DISP=OLD,VOL=SER=000001,
//          DCB=EROPT=SKF,LABEL=(,NL)
//          DD DSN=HB,UNIT=AFF=TAPEIN,LABEL=(,NL),VOL=SER=000002
//          DD DSN=HB,UNIT=AFF=TAPEIN,LABEL=(,NL),VOL=SER=000003
//          DD DSN=HB,UNIT=AFF=TAPEIN,LABEL=(,NL),VOL=SER=000004
//          DD DSN=HB,UNIT=AFF=TAPEIN,LABEL=(,NL),VOL=SER=000005
//          DD DSN=HB,UNIT=AFF=TAPEIN,LABEL=(,NL),VOL=SER=000006
//          DD DSN=HB,UNIT=AFF=TAPEIN,LABEL=(,NL),VOL=SER=000007
//          DD DSN=HB,UNIT=AFF=TAPEIN,LABEL=(,NL),VOL=SER=000008
//GO.TAPEOUT DD DSN=CHOI13,UNIT=TAPE,DISP=NEW
//SYSOUT DD SYSOUT=A
//
SVI00560
SVI00570
SVI00580
SVI00590
SVI00600
SVI00610
SVI00620
SVI00630
SVI00640
SVI00650
SVI00660
SVI00670
SVI00680
SVI00690
SVI00700
SVI00710
SVI00720
```

```

//SVIWSM14 JOB TYPRUN=HOLD
/** DIVORCE TABLE SUMMARY ***
// EXEC PLIFCLG
//PLI.SYSIN DD *
* PROCESS GS,NEST,OPT(TIME);
  CHOI: PROC OPTIONS(MAIN);
        TAB14=0;
        TAB15=0;
        TAB16=0;
        TAB17=0;
        OPEN FILE(TAPEIN) INPUT,
        FILE(TAPEOUT) OUTPUT,
        FILE(SYSPRINT) LINESIZE(132) PAGESIZE(60);
        ON ENDFILE(TAPEIN) END_RECORDS = 'YES';
        END_RECORDS = 'NO';
/* READ PART */
  READ FILE(TAPEIN) INTO(T_R);
  CALL @GECOM;
  DO WHILE (END_RECORDS = 'NO');
    CALL SET_RTN;
    READ FILE(TAPEIN) INTO(T_R);
    CALL @GECOM;
  END;
  CALL SUM_RTN;
  CALL WRITE_RTN;
  CLOSE FILE(TAPEIN),FILE(TAPEOUT);
/* DECLARE PART */
DCL TAPEIN FILE RECORD INPUT;
DCL 1 T_R,
    3 F1 CHAR(20),
    3 MDO PIC '99',
    3 GUN PIC '99',
    3 FDO PIC '99',
    3 FGUN CHAR(2),
    3 MSY PIC '99',
    3 MSW PIC '99',
    3 FSY PIC '99',
    3 FSW PIC '99',
    3 F2 CHAR(4),
    3 YR PIC '99',
    3 WOL PIC '99',
    3 F3 CHAR(1),
    3 DOVR PIC '9',
    3 F4 CHAR(2),
    3 MAG PIC '99',
    3 FAG PIC '99',
    3 F6 CHAR(28);
DCL TAPEOUT FILE RECORD OUTPUT;
DCL 1 TA_R,
    3 PYC PIC '99',
    3 PYR PIC '99',
    3 PJUL PIC '999',
    3 PCAN PIC '99',
    3 TAA(40) PIC '(8)9',
DCL YEAR(10) PIC '99' INIT(82,81,80,79,78,77,76,75,74,73);

```

SVI00010  
SVI00020  
SVI00030  
SVI00040  
SVI00050  
SVI00060  
SVI00070  
SVI00080  
SVI00090  
SVI00100  
SVI00110  
SVI00120  
SVI00130  
SVI00140  
SVI00150  
SVI00160  
SVI00170  
SVI00180  
SVI00190  
SVI00200  
SVI00210  
SVI00220  
SVI00230  
SVI00240  
SVI00250  
SVI00260  
SVI00270  
SVI00280  
SVI00290  
SVI00300  
SVI00310  
SVI00320  
SVI00330  
SVI00340  
SVI00350  
SVI00360  
SVI00370  
SVI00380  
SVI00390  
SVI00400  
SVI00410  
SVI00420  
SVI00430  
SVI00440  
SVI00450  
SVI00460  
SVI00470  
SVI00480  
SVI00490  
SVI00500  
SVI00510  
SVI00520  
SVI00530  
SVI00540  
SVI00550

```

DCL AGE151(30) PIC '99' INIT(00,15,20,25,30,35,36,37,38,39,40,41,42,
43,44,45,46,47,48,49,50,51,52,53,54,55,
60,65,70,75); SVI00560
SVI00570
SVI00580
DCL AGE152(38) PIC '99' INIT(00,15,20,21,22,23,24,25,26,27,
28,29,30,31,32,33,34,35,36,37,
38,39,40,41,42,43,44,45,46,47,
48,49,50,55,60,65,70,75); SVI00590
SVI00600
SVI00610
SVI00620
DCL AGE16(63) PIC '99' INIT(00,15,16,17,18,19,20,21,22,23,24,25,26,
27,28,29,30,31,32,33,34,35,36,37,38,39,
40,41,42,43,44,45,46,47,48,49,50,51,52,
53,54,55,56,57,58,59,60,61,62,63,64,65,
66,67,68,69,70,71,72,73,74,75,76); SVI00630
SVI00640
SVI00650
SVI00660
SVI00670
DCL DOD(14) PIC '99' INIT(0,11,21,22,23,31,32,33,34,35,36,37,38,39); SVI00680
DCL TAB14(10,13,15) FIXED DEC(8); SVI00690
DCL TAB15(10,32,40) FIXED DEC(8); SVI00700
DCL TAB16(10,65,15,2) FIXED DEC(8); SVI00710
DCL TAB17(10,7,15) FIXED DEC(8); SVI00720
DCL END_RECORDS CHAR(3); SVI00730
/* SUB PROCEDURE */ SVI00740
SET_RTN: PROCEDURE; SVI00750
DO N=1 BY 1 UNTIL(YR = YEAR(N) | N > 10); END; SVI00760
IF N > 10 THEN GOTO SET6; SVI00770
I=N; SVI00780
/* WOL PROCEDURE */ SVI00790
DO N = 1 BY 1 UNTIL(WOL = N | N > 11); END; SVI00800
J14=N+1; SVI00810
SVI00820
DO N=2 BY 1 UNTIL(MDO = DOD(N) | N > 14); END; SVI00830
K14=N; SVI00840
SVI00850
DO N=2 BY 1 UNTIL(MAG < AGE151(N) | N > 30); END; SVI00860
J15=N; SVI00870
IF J15 = 2 THEN J15 = 3; SVI00880
SVI00890
DO N = 2 BY 1 UNTIL(FAG < AGE152(N) | N > 38); END; SVI00900
K15=N; SVI00910
IF K15 = 2 THEN K15 = 3; SVI00920
SVI00930
DO N = 2 BY 1 UNTIL(MAG < AGE16(N) | N > 63); END; SVI00940
J161 = N; SVI00950
IF J161 = 2 THEN J161 = 4; SVI00960
DO N = 2 BY 1 UNTIL(FAG < AGE16(N) | N > 63); END; SVI00970
J162=N; SVI00980
IF J162 = 2 THEN J162 = 4; SVI00990
K161=K14; SVI01000
DO N = 2 BY 1 UNTIL(FDO = DOD(N) | N > 14); END; SVI01010
K162=N; SVI01020
J17=DOVR+1; SVI01030
K17=K14; SVI01040
SVI01050
TAB14(I,J14,K14) = TAB14(I,J14,K14)+1; SVI01060
TAB15(I,J15,K15) = TAB15(I,J15,K15)+1; SVI01070
TAB15(I,1,K15) = TAB15(I,1,K15)+1; SVI01080
TAB16(I,J161,K161,1) = TAB16(I,J161,K161,1)+1; SVI01090
TAB16(I,1,K161,1) = TAB16(I,1,K161,1)+1; SVI01100
TAB16(I,J162,K162,2) = TAB16(I,J162,K162,2)+1;

```



```

TAB16(I,1,K162,2) = TAB16(I,1,K162,2)+1;
TAB17(I,J17,K17) = TAB17(I,J17,K17)+1;
SET6:
SUM_RTN: PROC;
DO I=2 TO 13;
    TAB14(*,1,*) = TAB14(*,1,*) + TAB14(*,I,*);
END;
DO I=2 TO 15;
    TAB14(*,*,1)=TAB14(*,*,1)+TAB14(*,*,I);
END;
DO I=2 TO 40;
    TAB15(*,*,1) = TAB15(*,*,1) + TAB15(*,*,I);
END;
DO I=2 TO 15;
    TAB16(*,*,1,1)=TAB16(*,*,1,1)+TAB16(*,*,I,1);
    TAB16(*,*,1,2)=TAB16(*,*,1,2)+TAB16(*,*,I,2);
END;
DO I =1 TO 7;
    TAB17(*,1,*)=TAB17(*,1,*)+ TAB17(*,I,*);
END;
DO I=2 TO 15;
    TAB17(*,*,1) = TAB17(*,*,1) + TAB17(*,*,I);
END;
END SUM_RTN;
WRITE_RTN: PROCEDURE ;
CALL SUBWR (TAB14,14,10,13,15);
CALL SUBWR (TAB15,15,10,32,40);
N=0;
DO I=1 TO 10;
    DO J=1 TO 65;
        PYO=16;
        PYR=I;
        PJUL=J;
        DO K=1 TO 15;
            DO L=1 TO 2;
                PCAN=K;
                N=N+1;
                TAA(N)=TAB16(I,J,K,L);
            END;
        END;
        WRITE FILE(TAPEOUT) FROM (TA_R);
        N=0;
    END;
END;
END WRITE_RTN;
SUBWR: PROC (ARRAY,PP,II,JJ,KK);
DCL ARRAY(*,*,*) FIXED DEC(8);
DO I= 1 TO II;
    DO J=1 TO JJ;
        PYO=PP;
        PYR=I;
        PJUL=J;
        DO K=1 TO KK;
            TAA(K) = ARRAY(I,J,K);
        END;
    END;
END;

```

SVI01110  
SVI01120  
SVI01130  
SVI01140  
SVI01150  
SVI01160  
SVI01170  
SVI01180  
SVI01190  
SVI01200  
SVI01210  
SVI01220  
SVI01230  
SVI01240  
SVI01250  
SVI01260  
SVI01270  
SVI01280  
SVI01290  
SVI01300  
SVI01310  
SVI01320  
SVI01330  
SVI01340  
SVI01350  
SVI01360  
SVI01370  
SVI01380  
SVI01390  
SVI01400  
SVI01410  
SVI01420  
SVI01430  
SVI01440  
SVI01450  
SVI01460  
SVI01470  
SVI01480  
SVI01490  
SVI01500  
SVI01510  
SVI01520  
SVI01530  
SVI01540  
SVI01550  
SVI01560  
SVI01570  
SVI01580  
SVI01590  
SVI01600  
SVI01610  
SVI01620  
SVI01630  
SVI01640  
SVI01650

```

                END;
                WRITE FILE(TAPEOUT) FROM(TA_R);
            END;
        END;
    END SUBWP;
AGECOM: PROC;
    IF MSW > WOL THEN MAG = YR - MSY -1;
        ELSE MAG = YR - MSY;
    IF FSW > WOL THEN FAG = YR - FSY -1;
        ELSE FAG = YR - FSY;
END AGECOM;
END CHOI;
/*
//GO.TAPEIN DD UNIT=TAPE,DISP=(OLD,KEEP),VOL=SER=CHOI13,
//          DSN=CHOIN1,LABEL=(2,BLP),
//          DCB=(LRECL=80,RECFM=FB,BLKSIZE=3200,EROPT=SKP)
//GO.TAPEOUT DD UNIT=TAPE,DISP=(NEW,KEEP),DSN=CHOIVI,
//          DCB=(LRECL=329,RECFM=FB,BLKSIZE=3290,EROPT=SKP)
//GO.SYSPRINT DD SYSOUT=A
//GO.SYSOUT DD SYSOUT=A
//
SVI01660
SVI01670
SVI01680
SVI01690
SVI01700
SVI01710
SVI01720
SVI01730
SVI01740
SVI01750
SVI01760
SVI01770
SVI01780
SVI01790
SVI01800
SVI01810
SVI01820
SVI01830
SVI01840
SVI01850
SVI01860

```

```

//SVIWDL14 JOB CLASS=S, TYPRUN=HOLD
//*      DIVORCE  TABLE  SUMMARY  TO TAPE SUM ADD UP  *****
//      EXEC PLIFCLG
//PLI.SYSIN DD *
* PROCESS GS, NEST, OPT(TIME);
  CHOI: PROC OPTIONS(MAIN);
    TAB14=0;
    TAB15=0;
    TAB16=0;
    TAB17=0;
    OPEN FILE(TAPEIN1) INPUT,
           FILE(TAPEIN2) INPUT,
           FILE(TAPEOUT) OUTPUT,
           FILE(SYSPRINT) LINESIZE(132) PAGESIZE(60);
    ON ENDFILE(TAPEIN1) END1_REC='YES';
    ON ENDFILE(TAPEIN2) END2_REC='YES';
    END1_REC = 'NO';
    END2_REC = 'NO';
/* TAPE READ PART */
  READ FILE(TAPEIN1) INTO(T_R);
  DO WHILE(END1_REC = 'NO');
    IF GU = 14 THEN CALL SET14_R;
    ELSE
      IF GU = 15 THEN CALL SET15_R;
      ELSE
        IF GU = 16 THEN CALL SET16_R;
        ELSE CALL SET17_R;
        READ FILE(TAPEIN1) INTO(T_R);
      END;
    J=0;
    READ FILE(TAPEIN2) INTO(TI_R);
    DO WHILE(END2_REC = 'NO');
      IF KU = 14 THEN CALL SET14_RTN;
      ELSE
        IF KU = 15 THEN CALL SET15_RTN;
        ELSE
          IF KU = 16 THEN CALL SET16_RTN;
          ELSE CALL SET17_RTN;
          READ FILE(TAPEIN2) INTO(TI_R);
        END;
      CALL WRITE_RTN;
    CLOSE FILE(TAPEIN1), FILE(TAPEIN2), FILE(TAPEOUT);
/* DECLARE PART */
  DCL TAPEIN1 FILE RECORD INPUT;
  DCL 1 T_R,
      3 GU      PIC '99',
      3 GU1     PIC '99',
      3 GU2     PIC '999',
      3 GU3     PIC '99',
      3 AA(40)  PIC '(B)9';
  DCL TAPEIN2 FILE RECORD INPUT;
  DCL 1 TI_R,
      3 KU      PIC '99',
      3 KU1     PIC '99',
      3 KU2     PIC '999',
      3 KU3     PIC '99',

```

SVI00010  
SVI00020  
SVI00030  
SVI00040  
SVI00050  
SVI00060  
SVI00070  
SVI00080  
SVI00090  
SVI00100  
SVI00110  
SVI00120  
SVI00130  
SVI00140  
SVI00150  
SVI00160  
SVI00170  
SVI00180  
SVI00190  
SVI00200  
SVI00210  
SVI00220  
SVI00230  
SVI00240  
SVI00250  
SVI00260  
SVI00270  
SVI00280  
SVI00290  
SVI00300  
SVI00310  
SVI00320  
SVI00330  
SVI00340  
SVI00350  
SVI00360  
SVI00370  
SVI00380  
SVI00390  
SVI00400  
SVI00410  
SVI00420  
SVI00430  
SVI00440  
SVI00450  
SVI00460  
SVI00470  
SVI00480  
SVI00490  
SVI00500  
SVI00510  
SVI00520  
SVI00530  
SVI00540  
SVI00550

```

        3 BB(40)    PIC '(8)9';
DCL TAPEOUT FILE RECORD OUTPUT;
DCL 1 TA_R,
    3 GUA    PIC '99',
    3 GUB    PIC '99',
    3 GUC    PIC '999',
    3 GUD    PIC '99',
    3 CC(40) PIC '(8)9';
DCL TAB14(10,13,15) FIXED DEC(8);
DCL TAB15(10,32,40) FIXED DEC(8);
DCL TAB16(10,65,30) FIXED DEC(8);
DCL TAB17(10,7,15)  FIXED DEC(8);
DCL END1_REC CHAR(3);
DCL END2_REC CHAR(3);
/**** SUB PROCEDURE ****/
SET14_R: PROC;
    IF GU1 = 10 THEN GOTO SET14_E;
    I=GU1+1;
    J=GU2;
    DO K=1 TO 15;
        TAB14(I,J,K) = TAB14(I,J,K) + AA(K);
    END;
SET14_E: END SET14_R;
SET15_R: PROC;
    IF GU1=10 THEN GOTO SET15_E;
    I=GU1+1;
    J=GU2;
    DO K=1 TO 40;
        TAB15(I,J,K) = TAB15(I,J,K) + AA(K);
    END;
SET15_E: END SET15_R;
SET16_R: PROC;
    IF GU1 = 10 THEN GOTO SET16_E;
    I=GU1+1;
    J=GU2;
    DO K=1 TO 30;
        TAB16(I,J,K) = TAB16(I,J,K) + AA(K);
    END;
SET16_E: END SET16_R;
SET17_R: PROC;
    IF GU1 = 10 THEN GOTO SET17_E;
    I=GU1+1;
    J=GU2;
    DO K=1 TO 15;
        TAB17(I,J,K) = TAB17(I,J,K) + AA(K);
    END;
SET17_E: END SET17_R;
SET14_RTN: PROC;
    I=KU1;
    J=KU2;
    DO K=1 TO 15;
        TAB14(I,J,K)=TAB14(I,J,K) + BB(K);
    END;
END SET14_RTN;
SET15_RTN: PROC;

```

SVI00560  
 SVI00570  
 SVI00580  
 SVI00590  
 SVI00600  
 SVI00610  
 SVI00620  
 SVI00630  
 SVI00640  
 SVI00650  
 SVI00660  
 SVI00670  
 SVI00680  
 SVI00690  
 SVI00700  
 SVI00710  
 SVI00720  
 SVI00730  
 SVI00740  
 SVI00750  
 SVI00760  
 SVI00770  
 SVI00780  
 SVI00790  
 SVI00800  
 SVI00810  
 SVI00820  
 SVI00830  
 SVI00840  
 SVI00850  
 SVI00860  
 SVI00870  
 SVI00880  
 SVI00890  
 SVI00900  
 SVI00910  
 SVI00920  
 SVI00930  
 SVI00940  
 SVI00950  
 SVI00960  
 SVI00970  
 SVI00980  
 SVI00990  
 SVI01000  
 SVI01010  
 SVI01020  
 SVI01030  
 SVI01040  
 SVI01050  
 SVI01060  
 SVI01070  
 SVI01080  
 SVI01090  
 SVI01100

```

        I=KU1;
        J=KU2;
        DO K=1 TO 40;
            TAB15(I,J,K)=TAB15(I,J,K) + BB(K);
        END;
    END SET15_RTN;
    SET16_RTN: PROC;
        I=KU1;
        J=KU2;
        DO K=1 TO 30;
            TAB16(I,J,K)= TAB16(I,J,K) + BB(K);
        END;
    END SET16_RTN;
    SET17_RTN: PROC;
        I=KU1;
        J=KU2;
        DO K=1 TO 15;
            TAB17(I,J,K)=TAB17(I,J,K) + BB(K);
        END;
    END SET17_RTN;
    WRITE_RTN: PROC;
        CALL SUBWR(TAB14,10,13,15,14) ;
        CALL SUBWR(TAB15,10,32,40,15) ;
        CALL SUBWR(TAB16,10,65,30,16) ;
        CALL SUBWR(TAB17,10,7,15,17) ;
    END WRITE_RTN;
    SUBWR: PROC (ARRAY, II, JJ, KK, PP);
    DCL ARRAY(*,*,*) FIXED DEC(8);
    DO I=1 TO II;
        DO J=1 TO JJ;
            GUA=PP;
            GUB=I;
            GUC=J;
            DO K=1 TO KK;
                CC(K) = ARRAY(I,J,K);
            END;
            WRITE FILE(TAPEOUT) FROM(TA_R);
        END;
    END;
    END SUBWR;
END CHOI;
/*
//GO.TAPEIN1 DD DSN=SVIDWED,UNIT=TAPE,DISP=(OLD,KEEP),VOL=SER=UUPPAK,
//          DCB=(RECFM=FB,LRECL=329,BLKSIZE=3290,EROPT=SKP),
//          LABEL=(2,BLP)
//GO.TAPEIN2 DD DSN=SVIDWET,UNIT=TAPE,DISP=(OLD,KEEP),VOL=SER=AUPPAK,
//          DCB=(RECFM=FB,LRECL=329,BLKSIZE=3290,EROPT=SKP),
//          LABEL=(2,BLP)
//GO.TAPEOUT DD DSN=SVIDWMT,UNIT=TAPE,DISP=(NEW,KEEP),
//          DCB=(RECFM=FB,LRECL=329,BLKSIZE=3290,EROPT=SKP)
//GO.SYSPRINT DD SYSOUT=A
//GO.SYSOUT DD SYSOUT=A
//
SVI01110
SVI01120
SVI01130
SVI01140
SVI01150
SVI01160
SVI01170
SVI01180
SVI01190
SVI01200
SVI01210
SVI01220
SVI01230
SVI01240
SVI01250
SVI01260
SVI01270
SVI01280
SVI01290
SVI01300
SVI01310
SVI01320
SVI01330
SVI01340
SVI01350
SVI01360
SVI01370
SVI01380
SVI01390
SVI01400
SVI01410
SVI01420
SVI01430
SVI01440
SVI01450
SVI01460
SVI01470
SVI01480
SVI01490
SVI01500
SVI01510
SVI01520
SVI01530
SVI01540
SVI01550
SVI01560
SVI01570
SVI01580
SVI01590
SVI01600
SVI01610
SVI01620
SVI01630

```

```

//SVIWT14 JOB MSGLEVEL=(1,1),CLASS=A,TYPRUN=HOLD SVI00010
//TABSM EXEC COBUCLG,PARM='NOSOURCE' SVI00020
/** * TAB 14,15,16,17 EDITING * SVI00030
//COB.SYSIN DD * SVI00040
    ID DIVISION. SVI00050
    PROGRAM-ID. DIVRED. SVI00060
    ENVIRONMENT DIVISION. SVI00070
    CONFIGURATION SECTION. SVI00080
    SPECIAL-NAMES. SVI00090
        CO1 IS CHOI SVI00100
        CSP IS DOUB. SVI00110
    INPUT-OUTPUT SECTION. SVI00120
        SELECT T-F ASSIGN UT-S-TAPEIN. SVI00130
        SELECT P-F ASSIGN UT-S-PROUT. SVI00140
    DATA DIVISION. SVI00150
    FILE SECTION. SVI00160
    FD T-F BLOCK CONTAINS 10 RECORDS SVI00170
        LABEL RECORD STANDARD. SVI00180
01 T-R. SVI00190
    03 GU PIC 99. SVI00200
    03 GU1 PIC 99. SVI00210
    03 GU2 PIC 999. SVI00220
    03 GU3 PIC 99. SVI00230
    03 AA OCCURS 40 PIC 9(8). SVI00240
    FD P-F LABEL RECORD OMITTED. SVI00250
01 P-R PIC X(132). SVI00260
    WORKING-STORAGE SECTION. SVI00270
    77 I PIC 9(3) VALUE 0. SVI00280
    77 J PIC 9(3) VALUE 0. SVI00290
    77 K PIC 9(3) VALUE 0. SVI00300
    77 L PIC 9(3) VALUE 0. SVI00310
    77 M PIC 9(3) VALUE 1. SVI00320
    77 N PIC 9(3) VALUE 2. SVI00330
    77 B1 PIC 9(3) VALUE 1. SVI00340
    77 B2 PIC 9(3) VALUE 2. SVI00350
    77 B3 PIC 9(3) VALUE 5. SVI00360
    77 B4 PIC 9(3) VALUE 0. SVI00370
    77 S15 PIC 9 VALUE 0. SVI00380
    77 S16 PIC 9 VALUE 0. SVI00390
    77 S17 PIC 9 VALUE 0. SVI00400
    77 CNT PIC 9(3) VALUE 0. SVI00410
    77 D1 PIC 99 VALUE 0. SVI00420
01 LIST14. SVI00430
    03 F PIC X(15) VALUE SPACE. SVI00440
    03 P14 OCCURS 13 PIC Z(7)9. SVI00450
    03 P141 PIC Z(6)9. SVI00460
    03 P142 PIC Z(5)9. SVI00470
01 LIST151 SVI00480
    03 F PIC X(44) VALUE SPACE. SVI00490
    03 P151 OCCURS 8 PIC Z(10)9. SVI00500
01 LIST152. SVI00510
    03 P152 OCCURS 12 PIC Z(10)9. SVI00520
01 LIST161. SVI00530
    03 F PIC X(18) VALUE SPACE. SVI00540
    03 PTM PIC Z(8)9. SVI00550

```

	03	PTF	PIC Z(8)9.	SVI00560
	03	P161	OCCURS 12 PIC Z(7)9.	SVI00570
01		LIST162.		SVI00580
	03	P162.	OCCURS 16 PIC Z(7)9.	SVI00590
01		LIST17.		SVI00600
	03	F	PIC X(15) VALUE SPACE.	SVI00610
	03	WHL17	PIC Z(7)9.	SVI00620
	03	P17	OCCURS 12 PIC Z(7)9.	SVI00630
	03	P171	PIC Z(6)9.	SVI00640
	03	P172	PIC Z(5)9.	SVI00650
01		TABL15.		SVI00660
	03	TAB15	OCCURS 32.	SVI00670
	05	TA15	OCCURS 40 PIC 9(8).	SVI00680
01		TABL16.		SVI00690
	03	TAB16	OCCURS 65.	SVI00700
	05	TA16	OCCURS 30 PIC 9(8).	SVI00710
01		BIGYO	PIC X(6) VALUE "020610".	SVI00720
01		BGYO	REDEFINES BIGYO.	SVI00730
	03	B13	OCCURS 3 PIC 99.	SVI00740
		PROCEDURE	DIVISION.	SVI00750
		OPEN	INPUT T-F.	SVI00760
		OPEN	OUTPUT P-F.	SVI00770
		BUSAN.		SVI00780
		MOVE	SPACE TO P-R.	SVI00790
		WRITE	P-R AFTER ADVANCING CHOI.	SVI00800
		MOVE	SPACE TO P-R.	SVI00810
		WRITE	P-R AFTER ADVANCING 3.	SVI00820
		MOVE	SPACE TO P-R.	SVI00830
		WRITE	P-R AFTER ADVANCING 3.	SVI00840
		MOVE	SPACE TO P-R.	SVI00850
		WRITE	P-R AFTER ADVANCING 3.	SVI00860
		MOVE	SPACE TO P-R.	SVI00870
		WRITE	P-R AFTER ADVANCING 2.	SVI00880
		MOVE	SPACE TO P-R.	SVI00890
		WRITE	P-R AFTER ADVANCING 2.	SVI00900
		MOVE	SPACE TO P-R.	SVI00910
		READ-RTN.		SVI00920
		READ	T-F AT END GO TO CLOSE-RTN.	SVI00930
		COMPUTE	D1 = GU - 13.	SVI00940
		GO TO	14T 15T 16T 17T DEPENDING ON D1.	SVI00950
14T.				SVI00960
		ADD	1 TO CNT.	SVI00970
		IF	CNT = 66 PERFORM BUSAN	SVI00980
			MOVE 1 TO CNT M.	SVI00990
		PERFORM	14A VARYING I FROM 1 BY 1 UNTIL I > 13.	SVI01000
		MOVE	AA(14) TO P141.	SVI01010
		MOVE	AA(15) TO P142.	SVI01020
		IF	CNT = M WRITE P-R FROM LIST14 AFTER ADVANCING 3	SVI01030
			WRITE P-R FROM LIST14 AFTER ADVANCING DOUB	SVI01040
			MOVE SPACE TO P-R	SVI01050
			COMPUTE M = M + 13	SVI01060
			GO TO READ-RTN.	SVI01070
		PERFORM	A14 VARYING I FROM 1 BY 1 UNTIL I > 3.	SVI01080
		WRITE	P-R FROM LIST14 AFTER ADVANCING 1.	SVI01090
		MOVE	SPACE TO P-R.	SVI01100

GO TO READ-RTN.		SVI01110
A14. IF GU2 = B13 (I) GO TO L14-RTN.		SVI01120
L14-RTN.		SVI01130
WRITE P-R FROM LIST14 AFTER ADVANCING 2.		SVI01140
MOVE SPACE TO P-R.		SVI01150
GO TO READ-RTN.		SVI01160
14A. MOVE AA (I) TO P14 (I).		SVI01170
15T.		SVI01180
15. IF GU4 NOT = B1 GO TO 15T-PRY.		SVI01190
RT. MOVE GU2 TO I.		SVI01200
PERFORM A15 VARYING J FROM 1 BY 1 UNTIL J > 40.		SVI01210
GO TO READ-RTN.		SVI01220
A15. MOVE AA(J) TO TA15 (I,J).		SVI01230
15T-PRY.		SVI01240
PERFORM BUSAN.		SVI01250
PERFORM 15A THRU EX VARYING I FROM 1 BY 1 UNTIL I > 32.		SVI01260
MOVE 2 TO B2.		SVI01270
PERFORM BUSAN.		SVI01280
PERFORM 15C THRU ET VARYING I FROM 1 BY 1 UNTIL I > 32.		SVI01290
22. MOVE 2 TO B2.		SVI01300
WRITE P-R AFTER ADVANCING CHOI.		SVI01310
MOVE SPACE TO P-R.		SVI01320
WRITE P-R AFTER ADVANCING 2.		SVI01330
MOVE SPACE TO P-R.		SVI01340
PP. PERFORM 15E THRU EJ VARYING I FROM 1 BY 1 UNTIL I > 32.		SVI01350
PERFORM 22.		SVI01360
PERFORM 15H THRU EK VARYING I FROM 1 BY 1 UNTIL I > 32.		SVI01370
MOVE 2 TO B2.		SVI01380
MOVE GU4 TO B1.		SVI01390
PERFORM BUSAN.		SVI01400
PA. GO TO RT.		SVI01410
15A.		SVI01420
PERFORM 15B VARYING J FROM 1 BY 1 UNTIL J > 8.		SVI01430
IF I = 1 WRITE P-R FROM LIST151 AFTER ADVANCING 2		SVI01440
WRITE P-R FROM LIST151 AFTER ADVANCING DOUB		SVI01450
MOVE SPACE TO P-R		SVI01460
GO TO EX.		SVI01470
IF I = B2 WRITE P-R FROM LIST151 AFTER ADVANCING 2		SVI01480
MOVE SPACE TO P-R		SVI01490
COMPUTE B2 = B2 + 3		SVI01500
GO TO EX.		SVI01510
WRITE P-R FROM LIST151 AFTER ADVANCING 1.		SVI01520
MOVE SPACE TO P-R.		SVI01530
EX. EXIT.		SVI01540
15B. MOVE TA15 (I,J) TO P151 (J).		SVI01550
15C. PERFORM 15D VARYING J FROM 9 BY 1 UNTIL J > 20.		SVI01560
IF I = 1 WRITE P-R FROM LIST152 AFTER ADVANCING 2		SVI01570
WRITE P-R FROM LIST152 AFTER ADVANCING DOUB		SVI01580
MOVE SPACE TO P-R		SVI01590
GO TO ET.		SVI01600
IF I = B2 WRITE P-R FROM LIST152 AFTER ADVANCING 2		SVI01610
MOVE SPACE TO P-R		SVI01620
COMPUTE B2 = B2 + 3		SVI01630
GO TO ET.		SVI01640
WRITE P-R FROM LIST152 AFTER ADVANCING 1.		SVI01650



MOVE SPACE TO P-R.	SVI01660
ET. EXIT.	SVI01670
15D. COMPUTE K = J - 8.	SVI01680
MOVE TA15 (I,J) TO P152 (K).	SVI01690
15E. PERFORM 15F VARYING J FROM 21 BY 1 UNTIL J > 28.	SVI01700
IF I = 1 WRITE P-R FROM LIST151 AFTER ADVANCING 2	SVI01710
WRITE P-R FROM LIST151 AFTER ADVANCING DOUB	SVI01720
MOVE SPACE TO P-R	SVI01730
GO TO EJ.	SVI01740
IF I = B2 WRITE P-R FROM LIST151 AFTER ADVANCING 2	SVI01750
MOVE SPACE TO P-R	SVI01760
COMPUTE B2 = B2 + 3	SVI01770
GO TO EJ.	SVI01780
WRITE P-R FROM LIST151 AFTER ADVANCING 1.	SVI01790
MOVE SPACE TO P-R.	SVI01800
EJ. EXIT.	SVI01810
15F. COMPUTE K = J - 20.	SVI01820
MOVE TA15 (I,J) TO P151 (K).	SVI01830
15H. PERFORM 15J VARYING J FROM 29 BY 1 UNTIL J > 40.	SVI01840
IF I = 1 WRITE P-R FROM LIST152 AFTER ADVANCING 2	SVI01850
WRITE P-R FROM LIST152 AFTER ADVANCING DOUB	SVI01860
MOVE SPACE TO P-R	SVI01870
GO TO EK.	SVI01880
IF I = B2 WRITE P-R FROM LIST152 AFTER ADVANCING 2	SVI01890
MOVE SPACE TO P-R	SVI01900
COMPUTE B2 = B2 + 3	SVI01910
GO TO EK.	SVI01920
WRITE P-R FROM LIST152 AFTER ADVANCING 1.	SVI01930
MOVE SPACE TO P-R.	SVI01940
EK. EXIT.	SVI01950
15J. COMPUTE K = J - 28.	SVI01960
MOVE TA15 (I,J) TO P152 (K).	SVI01970
16T.	SVI01980
IF S16 = 1 GO TO 16T-BG.	SVI01990
PERFORM 15T-PRT THRU PP.	SVI02000
MOVE 1 TO S16.	SVI02010
MOVE 1 TO B1.	SVI02020
16T-BG.	SVI02030
IF GU1 NOT = B1 GO TO 16T-PRT.	SVI02040
16T-RT.	SVI02050
MOVE GU2 TO I.	SVI02060
PERFORM A16 VARYING J FROM 1 BY 1 UNTIL J > 30.	SVI02070
GO TO READ-RTN.	SVI02080
A16. MOVE AA (J) TO TA16 (I,J).	SVI02090
16T-PRT.	SVI02100
PERFORM 16A THRU EE VARYING I FROM 1 BY 1 UNTIL I > 65.	SVI02110
MOVE 5 TO B3.	SVI02120
PERFORM BUSAN.	SVI02130
PERFORM 16C THRU EP VARYING I FROM 1 BY 1 UNTIL I > 65.	SVI02140
PERFORM BUSAN.	SVI02150
PB. MOVE 5 TO B3.	SVI02160
MOVE GU1 TO B1.	SVI02170
GO TO 16T-RT.	SVI02180
16A.	SVI02190
MOVE TA16 (I,1) TO PTM.	SVI02200

MOVE TA16 (I,2) TO PTF.	SVI02210
PERFORM 16B VARYING J FROM 3 BY 1 UNTIL J > 14.	SVI02220
IF I = 1 WRITE P-R FROM LIST161 AFTER ADVANCING 3	SVI02230
WRITE P-R FROM LIST161 AFTER ADVANCING DOUB	SVI02240
MOVE SPACE TO P-R	SVI02250
GO TO EE.	SVI02260
IF I = 2 WRITE P-R FROM LIST161 AFTER ADVANCING 3	SVI02270
MOVE SPACE TO P-R	SVI02280
GO TO EE.	SVI02290
IF I = B3 WRITE P-R FROM LIST161 AFTER ADVANCING 2	SVI02300
MOVE SPACE TO P-R	SVI02310
COMPUTE B3 = B3 + 3	SVI02320
GO TO EE.	SVI02330
WRITE P-R FROM LIST161 AFTER ADVANCING 1.	SVI02340
MOVE SPACE TO P-R.	SVI02350
EE. EXIT.	SVI02360
16B. COMPUTE K = J - 2.	SVI02370
MOVE TA16 (I,J) TO P161 (K).	SVI02380
16C.	SVI02390
PERFORM 16D VARYING J FROM 15 BY 1 UNTIL J > 30.	SVI02400
IF I = 1 WRITE P-R FROM LIST162 AFTER ADVANCING 3	SVI02410
WRITE P-R FROM LIST162 AFTER ADVANCING DOUB	SVI02420
MOVE SPACE TO P-R	SVI02430
GO TO EP.	SVI02440
IF I = 2 WRITE P-R FROM LIST162 AFTER ADVANCING 3	SVI02450
MOVE SPACE TO P-R	SVI02460
GO TO EP.	SVI02470
IF I = B3 WRITE P-R FROM LIST162 AFTER ADVANCING 2	SVI02480
MOVE SPACE TO P-R	SVI02490
COMPUTE B3 = B3 + 3	SVI02500
GO TO EP.	SVI02510
WRITE P-R FROM LIST162 AFTER ADVANCING 1.	SVI02520
MOVE SPACE TO P-R.	SVI02530
EP. EXIT.	SVI02540
16D. COMPUTE K = J - 14.	SVI02550
MOVE TA16 (I,J) TO P162(K).	SVI02560
17T.	SVI02570
IF S17 = 1 GO TO 17T-BG.	SVI02580
PERFORM 16T-PRT.	SVI02590
MOVE 1 TO S17 B1.	SVI02600
MOVE 2 TO B2. MOVE 0 TO CNT.	SVI02610
17T-BG.	SVI02620
ADD 1 TO CNT. MOVE AA (1) TO WHL17.	SVI02630
PERFORM 17A VARYING I FROM 2 BY 1 UNTIL I > 13.	SVI02640
MOVE AA(14) TO P171.	SVI02650
MOVE AA(15) TO P172.	SVI02660
IF CNT = B1 WRITE P-R FROM LIST17 AFTER ADVANCING 2	SVI02670
WRITE P-R FROM LIST17 AFTER ADVANCING DOUB	SVI02680
MOVE SPACE TO P-R	SVI02690
COMPUTE B1 = B1 + 7	SVI02700
GO TO READ-RTN.	SVI02710
IF CNT = B2 WRITE P-R FROM LIST17 AFTER ADVANCING 2	SVI02720
MOVE SPACE TO P-R	SVI02730
COMPUTE B2 = B2 + 7	SVI02740
GO TO READ-RTN.	SVI02750

```
WRITE P-R FROM LIST17 AFTER ADVANCING 1.
MOVE SPACE TO P-R.
GO TO READ-RTN.
17A. COMPUTE J = I - 1.
MOVE AA (I) TO P17 (J).
CLOSE-RTN.
CLOSE T-F P-F.
STOP RUN.

/*
//GO.TAPEIN DD UNIT=TAPE,DSN=VISM59,DISP=(OLD,KEEP),VOL=SER=VI0001,
// LABEL=(2,BLF)
//SYSOUT DD SYSOUT=A
//GO.PROUT DD SYSOUT=A
//SYSDOUT DD SYSOUT=A
//
SVI02760
SVI02770
SVI02780
SVI02790
SVI02800
SVI02810
SVI02820
SVI02830
SVI02840
SVI02850
SVI02860
SVI02870
SVI02880
SVI02890
SVI02900
```

```

//SVDWST14 JOB CLASS=A, TYPRUN=HOLD SVD00010
//SELECT EXEC COBUCLG, PARM='CLIST, STATE, FLOW, NUM' SVD00020
//COB.SYSIN DD * SVD00030
    ID DIVISION. SVD00040
    PROGRAM-ID. MG14. SVD00050
    ENVIRONMENT DIVISION. SVD00060
    INPUT-OUTPUT SECTION. SVD00070
    FILE-CONTROL. SVD00080
        SELECT T-F ASSIGN UT-S-TAPEIN. SVD00090
        SELECT O-F ASSIGN UT-S-TAPEOUT. SVD00100
    DATA DIVISION. SVD00110
    FILE SECTION. SVD00120
    FD T-F LABEL RECORD STANDARD SVD00130
        BLOCK CONTAINS 40 RECORDS. SVD00140
    01 T-R. SVD00150
        03 F PIC X(33). SVD00160
        03 YR PIC XX. SVD00170
        03 F PIC X(45). SVD00180
    FD O-F LABEL RECORD STANDARD SVD00190
        BLOCK CONTAINS 40 RECORDS. SVD00200
    01 O-R PIC X(80). SVD00210
    WORKING-STORAGE SECTION. SVD00220
    77 T-CNT PIC 9(10) VALUE ZEROS. SVD00230
    77 O-CNT PIC 9(10) VALUE ZEROS. SVD00240
    PROCEDURE DIVISION. SVD00250
    SIJAK. SVD00260
        OPEN INPUT T-F OUTPUT O-F. SVD00270
    READ-T. SVD00280
        READ T-F AT END GO TO END-R. SVD00290
        ADD 1 TO T-CNT. SVD00300
        IF YR = '82' SVD00310
            WRITE O-R FROM T-R SVD00320
            ADD 1 TO O-CNT. SVD00330
        GO TO READ-T. SVD00340
    END-R. SVD00350
        CLOSE T-F O-F. SVD00360
        DISPLAY " INPUT COUNT " T-CNT. SVD00370
        DISPLAY "OUTPUT COUNT " O-CNT. SVD00380
        STOP RUN. SVD00390
/* SVD00400
//GO.TAPEIN DD DSN=HA, UNIT=TAPE, DISP=(OLD, KEEP), LABEL=(2, BLP), SVD00410
// DCB=(LRECL=80, BLKSIZE=3200, RECFM=FB, EROPT=SKP), VOL=SER=000001 SVD00420
//GO.TAPEOUT DD DSN=HO, UNIT=TAPE, DISP=(NEW, KEEP), VOL=(, , 99, SVD00430
// SER=SCRCH), DCB=(LRECL=80, BLKSIZE=3200, RECFM=FB, EROPT=SKP) SVD00440
//GO.SYSPRINT DD SYSOUT=A SVD00450
//GO.SYSDBOUT DD SYSOUT=A SVD00460
//GO.SYSOUT DD SYSOUT=A SVD00470
// SVD00480

```

```

//SVDUSR14 JOB CLASS=A, TYPRUN=HOLD                                SVD00010
//RANK EXEC PLIFCLG                                                SVD00020
//PLI,SYCIN DD *                                                  SVD00030
* PROCESS GS,NEST,OPT(2) ;                                         SVD00040
/* 1. THE DEATH TOLL BY DIAGNOSIS                                  */ SVD00050
/* 2. THE CLASSIFIABILITY BY REGION & DIAGNOSIS                  */ SVD00060
/* 3. THE DEATH TOLL BY DEATH PLACE                              */ SVD00070
/* 11. THE RANK OF DEATH CAUSE BY SEXES & AGES                   */ SVD00080
/* 13. THE RANK OF DEATH CAUSE BY REGIONS                         */ SVD00090
GOLD: PROC OPTIONS(MAIN) ;                                         SVD00100
DCL TF FILE RECORD INPUT;                                         SVD00110
DCL 1 TR,                                                           SVD00120
    2 DKEY CHAR(16),                                               SVD00130
    2 DBDO CHAR(2),                                               SVD00140
    2 DBGUN CHAR(2),                                              SVD00150
    2 DJDO CHAR(2),                                               SVD00160
    2 DJGUN CHAR(2),                                              SVD00170
    2 DSEX PIC '(1)9',                                           SVD00180
    2 DBYR PIC '(3)9',                                           SVD00190
    2 DBMN PIC '(2)9',                                           SVD00200
    2 DBDYY PIC '(2)9',                                          SVD00210
    2 DDYR PIC '(3)9',                                           SVD00220
    2 DBMN PIC '(2)9',                                          SVD00230
    2 DDDYY PIC '(2)9',                                          SVD00240
    2 DPLS PIC '(1)9',                                           SVD00250
    2 DDOPT PIC '(2)9',                                          SVD00260
    2 DJD PIC '(1)9',                                             SVD00270
    2 DMST PIC '(1)9',                                           SVD00280
    2 DEDT PIC '(1)9',                                           SVD00290
    2 DEACA CHAR(9),                                             SVD00300
    2 AGY PIC '(3)9',                                             SVD00310
    2 AGW PIC '(2)9',                                             SVD00320
    2 AGD PIC '(2)9',                                             SVD00330
    2 B(5) PIC '(3)9',                                           SVD00340
    2 F1 CHAR(4),                                                SVD00350
DCL OF FILE RECORD OUTPUT;                                        SVD00360
DCL 1 RR,                                                           SVD00370
    2 PYO PIC '9',                                               SVD00380
    2 AGE PIC '99',                                              SVD00390
    2 SEX PIC '9',                                               SVD00400
    2 ICD PIC '999',                                             SVD00410
    2 AMT PIC '(6)9',                                           SVD00420
    2 RATIO PIC '(6)9V99',                                       SVD00430
DCL AR1(2,4) PIC '(6)9';                                         SVD00440
DCL AR2(19,4,3) PIC '(6)9';                                       SVD00450
DCL AR3(2,4,3,2) PIC '(6)9V99';                                       SVD00460
DCL AR4(61,21,3) PIC '(6)9';                                       SVD00470
DCL AR5(61,20,2) PIC '(6)9V99';                                       SVD00480
DCL H PIC '(2)9' INIT(0);                                       SVD00490
DCL I PIC '(2)9' INIT(0);                                       SVD00500
DCL J PIC '(2)9' INIT(0);                                       SVD00510
DCL K PIC '(2)9' INIT(0);                                       SVD00520
DCL L PIC '(2)9' INIT(0);                                       SVD00530
DCL M PIC '(2)9' INIT(0);                                       SVD00540
DCL TTL1(2) CHAR(20) INIT(' REGISTERED NUMBER',                SVD00550

```

	' CLASSIFIABLE NUMBER ');	SVD00560
DCL TTL2(22) CHAR(20) INIT(	DIAGNOSIS ' ,	SVD00570
	REGION ' ,	SVD00580
	TOTAL ' ,	SVD00590
	SIBU ' ,	SVD00600
	LARGE ' ,	SVD00610
	MEDIUM ' ,	SVD00620
	SMALL ' ,	SVD00630
	GUNBU ' ,	SVD00640
	11 SEOUL ' ,	SVD00650
	21 BUSAN ' ,	SVD00660
	22 DEAGUU ' ,	SVD00670
	23 INCHEON ' ,	SVD00680
	31 KYOUNGGI ' ,	SVD00690
	32 KANGWON ' ,	SVD00700
	33 CHUNGBUK ' ,	SVD00710
	34 CHUNGNAM ' ,	SVD00720
	35 JEONBUK ' ,	SVD00730
	36 JEONNAM ' ,	SVD00740
	37 KYOUNGUK ' ,	SVD00750
	38 KYOUNGNAM ' ,	SVD00760
	39 JEJU ' ,	SVD00770
	00 EXTRA ' );	SVD00780
DCL TTL3(9) CHAR(20) INIT(	REGION ' ,	SVD00790
	TOTAL ' ,	SVD00800
	SIBU ' ,	SVD00810
	GUNBU ' ,	SVD00820
	PLACE ' ,	SVD00830
	HOME ' ,	SVD00840
	HOSPITAL ' ,	SVD00850
	OTHERS ' ,	SVD00860
	TOTAL ' );	SVD00870
DCL MOT(64) CHAR(11) INIT(	001 - 999 ' ,	SVD00880
	001 - 139 ' ,	SVD00890
	001 - 009 ' ,	SVD00900
	010 - 018 ' ,	SVD00910
	033 ' ,	SVD00920
	036 ' ,	SVD00930
	037 ' ,	SVD00940
	037 ' ,	SVD00950
	038 ' ,	SVD00960
	050 ' ,	SVD00970
	055 ' ,	SVD00980
	084 ' ,	SVD00990
	140 - 208 ' ,	SVD01000
	151 ' ,	SVD01010
	153 ' ,	SVD01020
	154 ' ,	SVD01030
	162 ' ,	SVD01040
	174 ' ,	SVD01050
	180 ' ,	SVD01060
	204 - 208 ' ,	SVD01070
	250 ' ,	SVD01080
	261 ' ,	SVD01090
	262,263 ' ,	SVD01100

' 280 - 285 '	SVD01110
' 320 - 322 '	SVD01120
' 390 - 459 '	SVD01130
' 390 - 392 '	SVD01140
' 393 - 398 '	SVD01150
' 401 - 405 '	SVD01160
' 410 - 414 '	SVD01170
' 410	SVD01180
' 427.5	SVD01190
' 430 - 438 '	SVD01200
' 436	SVD01210
' 437.8	SVD01220
' 440	SVD01230
' 480 - 486 '	SVD01240
' 487	SVD01250
' 490 - 493 '	SVD01260
' 531 - 533 '	SVD01270
' 540 - 543 '	SVD01280
' 571	SVD01290
' 580 - 589 '	SVD01300
' 600	SVD01310
' 630 - 639 '	SVD01320
' 640 - 646 '	SVD01330
' 740 - 759 '	SVD01340
' 760 - 779 '	SVD01350
' 767	SVD01360
' 780 - 799 '	SVD01370
' 797	SVD01380
' 799	SVD01390
' 800 - 999 '	SVD01400
' 800 - 829 '	SVD01410
' 850 - 869 '	SVD01420
' 940 - 949 '	SVD01430
' 960 - 989 '	SVD01440
'E800 -E949 '	SVD01450
'E810 -E819 '	SVD01460
'E880 -E889 '	SVD01470
'E950 -E959 '	SVD01480
'E960 -E969 '	SVD01490
DCL MF CHAR(2);	SVD01500
DCL BUN CHAR(20);	SVD01510
DCL HJ CHAR(6);	SVD01520
DCL TCNT PIC '(6)9' INIT(0);	SVD01530
DCL OCNT PIC '(6)9' INIT(0);	SVD01540
DCL ICNT PIC '(6)9' INIT(0);	SVD01550
DCL DCNT PIC '(6)9' INIT(0);	SVD01560
OPEN FILE(TF),FILE(OF),FILE(SYSPRINT) LINESIZE(133) PAGESIZE(66);	SVD01570
ON ENDFILE(TF) GO TO GROSS;	SVD01580
AR1 = 0;	SVD01590
AR2 = 0;	SVD01600
AR3 = 0;	SVD01610
AR4 = 0;	SVD01620
AR5 = 0;	SVD01630
RR = 0;	SVD01640
RD: READ FILE(TF) INTO(TR);	SVD01650

```

TCNT = TCNT + 1 ;
IF DJD = 0 THEN DCNT = DCNT + 1 ;
AR1(1,DJD+1) = AR1(1,DJD+1) + 1 ;
IF SUBSTR(DEACA,1,3) = '000' THEN GOTO NEXT;
AR1(2,DJD+1) = AR1(2,DJD+1) + 1 ;
NEXT: IF DJGUN < 30 THEN
DO;
    IF DJDO = 11|DJDO = 21|DJDO = 22|DJDO = 23 THEN L = 3;
    ELSE IF (DJDO = 31 & (DJGUN=11|DJGUN=12|DJGUN=14|DJGUN=15))|
        (DJDO=11|DJGUN = '3311')|
        (DJDO = '34' & DJGUN < 11 )|
        (DJDO=11|DJGUN = '3511')|
        (DJDO = '36' & DJGUN < 12 )|
        (DJDO=11|DJGUN = '3711')|
        (DJDO = '38' & DJGUN < 14 ) THEN L = 4;
    ELSE L = 5;
    IF SUBSTR(DEACA,1,3) ^= '000' THEN
        AR2(L,DJD+1,2) = AR2(L,DJD+1,2) + 1;
    ELSE AR2(L,DJD+1,3) = AR2(L,DJD+1,3) + 1;
END;
ELSE DO;
    L = 6;
    IF SUBSTR(DEACA,1,3) ^= '000' THEN
        AR2(6,DJD+1,2) = AR2(6,DJD+1,2) + 1;
    ELSE AR2(6,DJD+1,3) = AR2(6,DJD+1,3) + 1;
    END;
    IF DJDO = '11' THEN M = 7;
    ELSE IF DJDO = '21' THEN M = 8;
    ELSE IF DJDO = '22' THEN M = 9;
    ELSE IF DJDO = '23' THEN M = 10;
    ELSE IF DJDO = '31' THEN M = 11;
    ELSE IF DJDO = '32' THEN M = 12;
    ELSE IF DJDO = '33' THEN M = 13;
    ELSE IF DJDO = '34' THEN M = 14;
    ELSE IF DJDO = '35' THEN M = 15;
    ELSE IF DJDO = '36' THEN M = 16;
    ELSE IF DJDO = '37' THEN M = 17;
    ELSE IF DJDO = '38' THEN M = 18;
    ELSE IF DJDO = '39' THEN M = 19;
    ELSE M = 20;
    IF SUBSTR(DEACA,1,3) ^= '000' THEN
        AR2(M,DJD+1,2) = AR2(M,DJD+1,2) + 1;
    ELSE AR2(M,DJD+1,3) = AR2(M,DJD+1,3) + 1;
    IF DJDO < 31 | DJGUN < 31 THEN J = 2;
    ELSE J = 3;
    AR3(1,DPLS,J,1) = AR3(1,DPLS,J,1) + 1;
    IF SUBSTR(DEACA,1,3) ^= '000' THEN
        AR3(2,DPLS,J,1) = AR3(2,DPLS,J,1) + 1;
    IF SUBSTR(DEACA,1,3) = '000' THEN GOTO RD;
    IF AGY < 1 THEN J = 2;
    ELSE IF AGY < 5 THEN J = 3;
    ELSE IF AGY < 10 THEN J = 4;
    ELSE IF AGY < 15 THEN J = 5;
    ELSE IF AGY < 20 THEN J = 6;
    ELSE IF AGY < 25 THEN J = 7;

```

SVD01660  
SVD01670  
SVD01680  
SVD01690  
SVD01700  
SVD01710  
SVD01720  
SVD01730  
SVD01740  
SVD01750  
SVD01760  
SVD01770  
SVD01780  
SVD01790  
SVD01800  
SVD01810  
SVD01820  
SVD01830  
SVD01840  
SVD01850  
SVD01860  
SVD01870  
SVD01880  
SVD01890  
SVD01900  
SVD01910  
SVD01920  
SVD01930  
SVD01940  
SVD01950  
SVD01960  
SVD01970  
SVD01980  
SVD01990  
SVD02000  
SVD02010  
SVD02020  
SVD02030  
SVD02040  
SVD02050  
SVD02060  
SVD02070  
SVD02080  
SVD02090  
SVD02100  
SVD02110  
SVD02120  
SVD02130  
SVD02140  
SVD02150  
SVD02160  
SVD02170  
SVD02180  
SVD02190  
SVD02200



```

ELSE IF AGY < 30 THEN J = 8;
ELSE IF AGY < 35 THEN J = 9;
ELSE IF AGY < 40 THEN J = 10;
ELSE IF AGY < 45 THEN J = 11;
ELSE IF AGY < 50 THEN J = 12;
ELSE IF AGY < 55 THEN J = 13;
ELSE IF AGY < 60 THEN J = 14;
ELSE IF AGY < 65 THEN J = 15;
ELSE IF AGY < 70 THEN J = 16;
ELSE IF AGY < 75 THEN J = 17;
ELSE IF AGY < 80 THEN J = 18;
ELSE IF AGY < 85 THEN J = 19;
ELSE J = 20;
IF DJD /= 1 THEN GOTO RD;
DO I = 1 TO 5;
  IF B(I) /= 0 THEN DO;
    AR4(B(I),J,DSEX+1) = AR4(B(I),J,DSEX+1) + 1;
    AR5(B(I),L,1) = AR5(B(I),L,1) + 1;
    AR5(B(I),M,1) = AR5(B(I),M,1) + 1;
  END;
END;
AR4(1,J,DSEX+1) = AR4(1,J,DSEX+1) + 1;
AR5(1,L,1) = AR5(1,L,1) + 1;
AR5(1,M,1) = AR5(1,M,1) + 1;
GOTO RD;
GROSS:
AR1(*,1) = AR1(*,2) + AR1(*,3) + AR1(*,4);
AR2(*,*,1) = AR2(*,*,2) + AR2(*,*,3);
AR2(*,1,*) = AR2(*,2,*) + AR2(*,3,*) + AR2(*,4,*);
AR2(2,*,*) = AR2(3,*,*) + AR2(4,*,*) + AR2(5,*,*);
AR2(1,*,*) = AR2(2,*,*) + AR2(6,*,*);
AR3(*,*,1,*) = AR3(*,*,2,*) + AR3(*,*,3,*);
AR3(*,4,*,*) = AR3(*,1,*,*) + AR3(*,2,*,*) + AR3(*,3,*,*);
DO H = 1 TO 2;
  DO I = 1 TO 4;
    DO J = 1 TO 3;
      IF AR3(H,4,J,1) = 0
      THEN AR3(H,I,J,2) = 0 ;
      ELSE AR3(H,I,J,2) = AR3(H,I,J,1) * 100 / AR3(H,4,J,1);
    END;
  END;
END;
AR4(*,*,1) = AR4(*,*,2) + AR4(*,*,3);
DO K = 2 TO 21;
  AR4(*,4,*) = AR4(*,1,*) + AR4(*,K,*);
END;
AR5(*,2,1) = AR5(*,3,1) + AR5(*,4,1) + AR5(*,5,1);
AR5(*,1,1) = AR5(*,2,1) + AR5(*,6,1);
DO I = 1 TO 61;
  DO J = 1 TO 20;
    IF AR5(1,J,1) /= 0 THEN DO;
      AR5(I,J,2) = AR5(I,J,1) * 100 / AR5(1,J,1); END;
    END;
  END;
END;
PRT:

```

SVD02210  
SVD02220  
SVD02230  
SVD02240  
SVD02250  
SVD02260  
SVD02270  
SVD02280  
SVD02290  
SVD02300  
SVD02310  
SVD02320  
SVD02330  
SVD02340  
SVD02350  
SVD02360  
SVD02370  
SVD02380  
SVD02390  
SVD02400  
SVD02410  
SVD02420  
SVD02430  
SVD02440  
SVD02450  
SVD02460  
SVD02470  
SVD02480  
SVD02490  
SVD02500  
SVD02510  
SVD02520  
SVD02530  
SVD02540  
SVD02550  
SVD02560  
SVD02570  
SVD02580  
SVD02590  
SVD02600  
SVD02610  
SVD02620  
SVD02630  
SVD02640  
SVD02650  
SVD02660  
SVD02670  
SVD02680  
SVD02690  
SVD02700  
SVD02710  
SVD02720  
SVD02730  
SVD02740  
SVD02750

```

PUT PAGE; SVD02760
PUT SKIP(3) EDIT((42)'*')(X(10),A); SVD02770
PUT SKIP(1) EDIT('* 1 THE DEATHES BY MEDICAL EXAMINATION *') SVD02780
(X(10),A); SVD02790
PUT SKIP(1) EDIT((42)'*')(X(10),A); SVD02800
PUT SKIP(3) EDIT(' TOTAL', SVD02810
' DOCTOR', SVD02820
' HERB DR.', SVD02830
' OTHERS') SVD02840
(X(30),4 A); SVD02850
PUT SKIP(1) EDIT((80)'=')(X(10),A); SVD02860
DO I = 1 TO 2; SVD02870
BUN = TTL1(I); SVD02880
PUT SKIP(3) EDIT(BUN,(AR1(I,J) DO J = 1 TO 4)) SVD02890
(X(10),A,4 F(15)); SVD02900
END; SVD02910
PUT PAGE; SVD02920
PUT SKIP(3) EDIT((42)'*')(X(8),A); SVD02930
PUT SKIP EDIT('* 2 THE CLASSIFIABILITY BY REGION & EX. *') SVD02940
(X(8),A); SVD02950
PUT SKIP EDIT((42)'*')(X(8),A); SVD02960
PUT SKIP EDIT(TTL2(1),' TOTAL', SVD02970
' DOCTOR', SVD02980
' HERB DOCTOR', SVD02990
' OTHERS') SVD03000
(X(8),A,4 A); SVD03010
PUT SKIP EDIT(TTL2(2),' TT PP IM', SVD03020
' TT PP IM', SVD03030
' TT PP IM', SVD03040
' TT PP IM') SVD03050
(X(8),A,4 A); SVD03060
PUT SKIP EDIT((124)'=')(X(8),A); SVD03070
DO I = 1 TO 19; SVD03080
IF I = 2|I = 3|I = 6|I = 7 THEN PUT SKIP(2); SVD03090
BUN = TTL2(I+2); SVD03100
PUT SKIP(2) EDIT(BUN,((AR2(I,J,K) DO K = 1 TO 3) SVD03110
DO J = 1 TO 4)) SVD03120
(X(8),A,4(X(2),3 F(8))); SVD03130
END; SVD03140
DO H = 1 TO 2; SVD03150
PUT PAGE; SVD03160
PUT SKIP(3) EDIT((42)'*')(X(10),A); SVD03170
PUT SKIP(1) EDIT('* 3 THE DEATH TOLL BY DEATH PLACE *') SVD03180
(X(10),A); SVD03190
PUT SKIP(1) EDIT((42)'*')(X(10),A); SVD03200
PUT SKIP EDIT((TTL3(K) DO K = 1 TO 4))(X(10),A,3(X(5),A)); SVD03210
PUT SKIP EDIT(TTL3(5),' AMT RATIO', SVD03220
' AMT RATIO', SVD03230
' AMT RATIO') SVD03240
(X(10),A,3(X(5),A)); SVD03250
PUT SKIP EDIT((95)'=')(X(10),A); SVD03260
DO I = 1 TO 4; SVD03270
BUN= TTL3(I+5); SVD03280
PUT SKIP(3) EDIT(BUN,((AR3(H,I,J,K) DO K = 1 TO 2) SVD03290
DO J = 1 TO 3)) SVD03300

```



```

//SVDWS714 JOB TYPRUN=HOLD
//* SUMMARY BY ICD(17) ***
//SUM17 EXEC COBUCLG
//COB.SYSIN DD *
    ID DIVISION.
    PROGRAM-ID. SUMMARY.
    ENVIRONMENT DIVISION.
    INPUT-OUTPUT SECTION.
    FILE-CONTROL.
        SELECT T-F ASSIGN UT-S-TAPEIN.
        SELECT TA-F ASSIGN UT-S-TAPEOUT.
    DATA DIVISION.
    FILE SECTION.
    FD T-F BLOCK CONTAINS 40 RECORDS
        LABEL RECORD STANDARD.
01 T-R.
    02 DBIGYO.
    03 F PIC X(3).
    03 WOL PIC 99.
    03 F PIC X(11).
    03 F PIC X(4).
    03 DO PIC 99.
    03 GUN PIC 99.
    03 SEX PIC 9.
    03 F PIC X(15).
    03 OP PIC 99.
    03 CU PIC 9.
    03 F PIC XX.
    03 WON.
        05 WO PIC 999.
            88 1T VALUE 001 THRU 139.
            88 2T VALUE 140 THRU 239.
            88 3T VALUE 240 THRU 279.
            88 4T VALUE 280 THRU 289.
            88 5T VALUE 290 THRU 319.
            88 6T VALUE 320 THRU 389.
            88 7T VALUE 390 THRU 459.
            88 8T VALUE 460 THRU 519.
            88 9T VALUE 520 THRU 579.
            88 10T VALUE 580 THRU 629.
            88 11T VALUE 630 THRU 679.
            88 12T VALUE 680 THRU 709.
            88 13T VALUE 710 THRU 739.
            88 14T VALUE 740 THRU 759.
            88 15T VALUE 760 THRU 779.
            88 16T VALUE 780 THRU 799.
            88 17T VALUE 800 THRU 999.
        05 F PIC X(6).
    03 WONS REDEFINES WON.
        05 WONS PIC XXXX.
        05 F PIC X(5).
    03 YEA PIC 999.
    03 WS PIC 99.
    03 ISS PIC 99.
    02 F PIC X(19).

```

SVD00010  
SVD00020  
SVD00030  
SVD00040  
SVD00050  
SVD00060  
SVD00070  
SVD00080  
SVD00090  
SVD00100  
SVD00110  
SVD00120  
SVD00130  
SVD00140  
SVD00150  
SVD00160  
SVD00170  
SVD00180  
SVD00190  
SVD00200  
SVD00210  
SVD00220  
SVD00230  
SVD00240  
SVD00250  
SVD00260  
SVD00270  
SVD00280  
SVD00290  
SVD00300  
SVD00310  
SVD00320  
SVD00330  
SVD00340  
SVD00350  
SVD00360  
SVD00370  
SVD00380  
SVD00390  
SVD00400  
SVD00410  
SVD00420  
SVD00430  
SVD00440  
SVD00450  
SVD00460  
SVD00470  
SVD00480  
SVD00490  
SVD00500  
SVD00510  
SVD00520  
SVD00530  
SVD00540  
SVD00550

FD TA-F BLOCK CONTAINS 40 RECORDS	SVD00560
LABEL RECORD STANDARD.	SVD00570
01 TA-R.	SVD00580
03 TTT          PIC X(61).	SVD00590
03 AA           PIC 999.	SVD00600
03 BB           PIC 999.	SVD00610
03 CC           PIC 999.	SVD00620
03 F            PIC X(10).	SVD00630
WORKING-STORAGE SECTION.	SVD00640
77 ECNT PIC 9(6) VALUE 0.	SVD00650
77 OCNT PIC 9(6) VALUE 0.	SVD00660
PROCEDURE DIVISION.	SVD00670
OPEN INPUT T-F.	SVD00680
OPEN OUTPUT TA-F.	SVD00690
MOVE SPACE TO TA-R.	SVD00700
READ-RTN.	SVD00710
READ T-F AT END GO TO END-RTN.	SVD00720
ADD 1 TO ECNT.	SVD00730
EXAMINE WON REPLACING ALL SPACE BY 0.	SVD00740
IF 1T          MOVE 1 TO AA.	SVD00750
IF 2T          MOVE 2 TO AA.	SVD00760
IF 3T          MOVE 3 TO AA.	SVD00770
IF 4T          MOVE 4 TO AA.	SVD00780
IF 5T          MOVE 5 TO AA.	SVD00790
IF 6T          MOVE 6 TO AA.	SVD00800
IF 7T          MOVE 7 TO AA.	SVD00810
IF 8T          MOVE 8 TO AA.	SVD00820
IF 9T          MOVE 9 TO AA.	SVD00830
IF 10T         MOVE 10 TO AA.	SVD00840
IF 11T         MOVE 11 TO AA.	SVD00850
IF 12T         MOVE 12 TO AA.	SVD00860
IF 13T         MOVE 13 TO AA.	SVD00870
IF 14T         MOVE 14 TO AA.	SVD00880
IF 15T         MOVE 15 TO AA.	SVD00890
IF 16T         MOVE 16 TO AA.	SVD00900
IF 17T         MOVE 17 TO AA.	SVD00910
MOVE DBIGYO TO TTT.	SVD00920
WRITE TA-R.	SVD00930
MOVE SPACE TO TA-R.	SVD00940
ADD 1 TO OCNT.	SVD00950
GO TO READ-RTN.	SVD00960
END-RTN.	SVD00970
DISPLAY " ECNT = " ECNT.	SVD00980
DISPLAY " OCNT = " OCNT.	SVD00990
CLOSE T-F TA-F.	SVD01000
OPEN INPUT TA-F.	SVD01010
RD-RTN.	SVD01020
READ TA-F AT END CLOSE TA-F STOP RUN.	SVD01030
DISPLAY " " TA-R.	SVD01040
PER-RTN.	SVD01050
PERFORM RD-RTN 700 TIMES.	SVD01060
CLOSE TA-F.	SVD01070
STOP RUN.	SVD01080
/*	SVD01090
//GO.TAPEIN DD DSN=SVBWED,UNIT=TAPE,DISP=OLD,VOL=SER=SNOW14,	SVD01100

FILE: SVDWS714 17SUM A1 VM/SP RELEASE 2.1 EXPRESS PUT8302+ SLU203

```
//          DCB=EROPT=SKP,LABEL=(2,BLP)          SVD01110
//GO.TAPEOUT DD DSN=SNOW11,UNIT=TAPE,DISP=(NEW,KEEP), SVD01120
//          VOL=(, ,99,SER=SCRCH)                SVD01130
//SYSOUT DD SYSOUT=A                             SVD01140
//                                                SVD01150
```

```

//SVDWT714 JOB CLASS=A, TYPRUN=HOLD
//TAB17 EXEC PLIFCLG
//PL1.SYSIN DD *
* PROCESS GS, NEST, OPT(2);
  GOLD: PROC OPTIONS(MAIN);
    DCL TF FILE RECORD INPUT;
    DCL 1 TR,
        2 DKEY CHAR(16),
        2 DBDO CHAR(2),
        2 DBGUN CHAR(2),
        2 DJDO CHAR(2),
        2 DJGUN CHAR(2),
        2 DSEX PIC '(1)9',
        2 DBYR PIC '(3)9',
        2 DBMN PIC '(2)9',
        2 DBDYY PIC '(2)9',
        2 DDYR PIC '(3)9',
        2 DDMN PIC '(2)9',
        2 DDDYY PIC '(2)9',
        2 DFPLS PIC '(1)9',
        2 DDOPT PIC '(2)9',
        2 DJD PIC '(1)9',
        2 DMST PIC '(1)9',
        2 DEDT PIC '(1)9',
        2 DEACA CHAR(9),
        2 AGY PIC '(3)9',
        2 AGW PIC '(2)9',
        2 AGD PIC '(2)9',
        2 BB PIC '(3)9',
        2 AA PIC '(3)9',
        2 CC CHAR(3),
        2 EAA CHAR(3),
        2 EBB CHAR(3),
        2 F1 CHAR(4),
    DCL AR1(2,18,3,4) PIC '(6)9';
    DCL AR2(2,3,18,3,14) PIC '(6)9';
    DCL AR3(2,18,3,7) PIC '(6)9';
    DCL AR4(2,18,14,3) PIC '(6)9';
    DCL AR5(18,3,2,6) PIC '(6)9';
    DCL AR6(18,3,3,5) PIC '(6)9';
    DCL AR7(2,18,3,2,6) PIC '(6)9';
    DCL HJ CHAR(6);
    DCL BUN CHAR(11);
    DCL MF CHAR(2);
    DCL TCNT PIC '(6)9' INIT(0);
    DCL OCNT PIC '(6)9' INIT(0);
    DCL ICNT PIC '(6)9' INIT(0);
    DCL H PIC '(1)9' INIT(0);
    DCL I PIC '(1)9' INIT(0);
    DCL J PIC '(2)9' INIT(0);
    DCL K PIC '(2)9' INIT(0);
    OPEN FILE(TF), FILE(SYSPRINT) LINESIZE(133) PAGESIZE(60);
    ON ENDFILE(TF) GO TO GROSS;
    ON ENDPAGE(SYSPRINT) BEGIN; PUT PAGE; PUT SKIP(5); END;
    AR1 = 0;

```

SVD00010  
SVD00020  
SVD00030  
SVD00040  
SVD00050  
SVD00060  
SVD00070  
SVD00080  
SVD00090  
SVD00100  
SVD00110  
SVD00120  
SVD00130  
SVD00140  
SVD00150  
SVD00160  
SVD00170  
SVD00180  
SVD00190  
SVD00200  
SVD00210  
SVD00220  
SVD00230  
SVD00240  
SVD00250  
SVD00260  
SVD00270  
SVD00280  
SVD00290  
SVD00300  
SVD00310  
SVD00320  
SVD00330  
SVD00340  
SVD00350  
SVD00360  
SVD00370  
SVD00380  
SVD00390  
SVD00400  
SVD00410  
SVD00420  
SVD00430  
SVD00440  
SVD00450  
SVD00460  
SVD00470  
SVD00480  
SVD00490  
SVD00500  
SVD00510  
SVD00520  
SVD00530  
SVD00540  
SVD00550

```

AR2 = 0; SVD00560
AR3 = 0; SVD00570
AR4 = 0; SVD00580
AR5 = 0; SVD00590
ARA = 0; SVD00600
AR7 = 0; SVD00610
RD: READ FILE(TF) INTO(TR); SVD00620
    TCNT = TCNT + 1; SVD00630
    IF SUBSTR(DEACA,1,3) = '000' THEN GOTO RD; SVD00640
/* 4 DEATH CAUSE BY DEATH PLACE */ SVD00650
    J = BB + 1; SVD00660
    K = DSEX + 1; SVD00670
    L = DPLS + 1; SVD00680
    IF DJD = '1' THEN AR1(2,J,K,L) = AR1(2,J,K,L) + 1; SVD00690
    AR1(1,J,K,L) = AR1(1,J,K,L) + 1; SVD00700
/* 10 AMOUNT OF DEATH BY SEXES & AGE & DEATH CAUSE */ SVD00710
/* 10'S INTERVAL */ SVD00720
    IF DJDO < 31 | DJGUN < 31 THEN I = 2; SVD00730
    ELSE I = 3; SVD00740
        IF AGY < 1 THEN L = 2; SVD00750
        ELSE IF AGY < 5 THEN L = 3; SVD00760
        ELSE IF AGY < 15 THEN L = 4; SVD00770
        ELSE IF AGY < 25 THEN L = 5; SVD00780
        ELSE IF AGY < 35 THEN L = 6; SVD00790
        ELSE IF AGY < 45 THEN L = 7; SVD00800
        ELSE IF AGY < 55 THEN L = 8; SVD00810
        ELSE IF AGY < 65 THEN L = 9; SVD00820
        ELSE IF AGY < 75 THEN L = 10; SVD00830
        ELSE L = 11; SVD00840
    IF DJD = '1' THEN DO; SVD00850
        AR2(2,I,BB+1,DSEX+1,L) = AR2(2,I,BB+1,DSEX+1,L) + 1; SVD00860
        AR2(2,I,BB+1,DSEX+1,L) = AR2(2,I,BB+1,DSEX+1,L) + 1; END; SVD00870
        AR2(1,I,BB+1,DSEX+1,L) = AR2(1,I,BB+1,DSEX+1,L) + 1; SVD00880
        AR2(1,I,BB+1,DSEX+1,L) = AR2(1,I,BB+1,DSEX+1,L) + 1; SVD00890
/* 10 AMOUNT OF DEATH BY SEXES & AGE & DEATH CAUSE */ SVD00900
/* 5 GROUP OF AGE */ SVD00910
        IF AGY < 1 THEN L = 2; SVD00920
        ELSE IF AGY < 15 THEN L = 3; SVD00930
        ELSE IF AGY < 45 THEN L = 4; SVD00940
        ELSE IF AGY < 65 THEN L = 5; SVD00950
        ELSE L = 6; SVD00960
        IF DJD = '1' THEN SVD00970
            AR3(2,BB+1,DSEX+1,L) = AR3(2,BB+1,DSEX+1,L) + 1; SVD00980
            AR3(1,BB+1,DSEX+1,L) = AR3(1,BB+1,DSEX+1,L) + 1; SVD00990
/* 7 DEATH CAUSE BY SEXES & MONTH */ SVD01000
            IF DJD = '1' THEN SVD01010
                AR4(2,BB+1,DDMN+1,DSEX+1) = AR4(2,BB+1,DDMN+1,DSEX+1) + 1; SVD01020
                AR4(1,BB+1,DDMN+1,DSEX+1) = AR4(1,BB+1,DDMN+1,DSEX+1) + 1; SVD01030
/* 5 AMOUNT OF DEATH BY DEATH CAUSE & EDUCATION */ SVD01040
                IF DJD = '1' THEN SVD01050
                    AR5(BB+1,DSEX+1,2,DEDT+1) = AR5(BB+1,DSEX+1,2,DEDT+1) + 1; SVD01060
                    AR5(BB+1,DSEX+1,1,DEDT+1) = AR5(BB+1,DSEX+1,1,DEDT+1) + 1; SVD01070
/* 8 DEATH CAUSE BY SEXES & REGION & OCCUPATION */ SVD01080
                IF DJDO < 31 | DJGUN < 31 THEN K = 2; SVD01090
                ELSE K = 3; SVD01100

```



```

        IF DOPT = 01 | DOPT = 02 | DOPT = 03 THEN L=1;          SVD01110
    ELSE IF DOPT = 04 | DOPT = 05 THEN L=2;                    SVD01120
    ELSE IF DOPT = 06 THEN L=3;                                SVD01130
    ELSE IF DOPT = 07 THEN L=4;                                SVD01140
    ELSE L=5;                                                  SVD01150
    AR6(BB+1,DSEX+1,1,L) = AR6(BB+1,DSEX+1,1,L) + 1;          SVD01160
    AR6(BB+1,DSEX+1,K,L) = AR6(BB+1,DSEX+1,K,L) + 1;          SVD01170
/* 6 DEATH CAUSE BY SEXES & MARITAL STATUS */                SVD01180
    IF DJD = '1' THEN                                          SVD01190
    AR7(2,BB+1,DSEX+1,2,DMST+1) = AR7(2,BB+1,DSEX+1,2,DMST+1) + 1; SVD01200
    AR7(1,BB+1,DSEX+1,1,DMST+1) = AR7(1,BB+1,DSEX+1,1,DMST+1) + 1; SVD01210
    GOTO RD;                                                  SVD01220
    SVD01230
GROSS:                                                       SVD01240
    AR1(*,*,*,1) = AR1(*,*,*,2) + AR1(*,*,*,3) + AR1(*,*,*,4); SVD01250
    AR1(*,*,1,*) = AR1(*,*,2,*) + AR1(*,*,3,*);             SVD01260
    DO J = 2 TO 18;                                           SVD01270
        AR1(*,1,*,*) = AR1(*,1,*,*) + AR1(*,J,*,*);        SVD01280
    END;                                                       SVD01290
    DO L = 2 TO 12;                                           SVD01300
        AR2(*,*,*,*,1) = AR2(*,*,*,*,1) + AR2(*,*,*,*,L); SVD01310
    END;                                                       SVD01320
    AR2(*,*,*,1,*) = AR2(*,*,*,2,*) + AR2(*,*,*,3,*);      SVD01330
    DO J = 2 TO 18;                                           SVD01340
        AR2(*,*,1,*,*) = AR2(*,*,1,*,*) + AR2(*,*,J,*,*); SVD01350
    END;                                                       SVD01360
    DO K = 2 TO 7;                                             SVD01370
        AR3(*,*,*,1) = AR3(*,*,*,1) + AR3(*,*,*,K);         SVD01380
    END;                                                       SVD01390
    AR3(*,*,1,*) = AR3(*,*,2,*) + AR3(*,*,3,*);             SVD01400
    DO L = 2 TO 18;                                           SVD01410
        AR3(*,1,*,*) = AR3(*,1,*,*) + AR3(*,L,*,*);         SVD01420
    END;                                                       SVD01430
    AR4(*,*,*,1) = AR4(*,*,*,2) + AR4(*,*,*,3);             SVD01440
    DO K = 2 TO 14;                                           SVD01450
        AR4(*,*,1,*) = AR4(*,*,1,*) + AR4(*,*,K,*);         SVD01460
    END;                                                       SVD01470
    DO J = 2 TO 18;                                           SVD01480
        AR4(*,1,*,*) = AR4(*,1,*,*) + AR4(*,J,*,*);         SVD01490
    END;                                                       SVD01500
    DO L = 2 TO 6;                                             SVD01510
        AR5(*,*,*,1) = AR5(*,*,*,1) + AR5(*,*,*,L);         SVD01520
    END;                                                       SVD01530
    AR5(*,1,*,*) = AR5(*,2,*,*) + AR5(*,3,*,*);             SVD01540
    DO L = 1 TO 18;                                           SVD01550
        AR5(1,*,*,*) = AR5(1,*,*,*) + AR5(L,*,*,*);         SVD01560
    END;                                                       SVD01570
    AR6(*,1,*,*) = AR6(*,2,*,*) + AR6(*,3,*,*);             SVD01580
    DO L = 2 TO 18;                                           SVD01590
        AR6(1,*,*,*) = AR6(1,*,*,*) + AR6(L,*,*,*);         SVD01600
    END;                                                       SVD01610
    AR7(*,*,*,*,1) = AR7(*,*,*,*,2)+AR7(*,*,*,*,3)+AR7(*,*,*,*,4) SVD01620
                    + AR7(*,*,*,*,5)+AR7(*,*,*,*,6);         SVD01630
    AR7(*,*,1,*,*) = AR7(*,*,2,*,*)+AR7(*,*,3,*,*);        SVD01640
    DO J = 2 TO 18;                                           SVD01650

```

```

        AR7(*,I,*,*,*) = AR7(*,I,*,*,*)+AR7(*,J,*,*,*);
    END;
PRT:
    PUT PAGE;
    PUT SKIP(5) EDIT((42)'*')(X(10),A);
    PUT SKIP(1) EDIT('* 4 DISTRIBUTION OF DEATH BY DEATH PLACE *')
        (X(10),A);
    PUT SKIP(1) EDIT((42)'*')(X(10),A);
    DO I = 1 TO 2;
        IF I = 2 THEN DO; PUT PAGE; PUT SKIP(5); END;
        PUT SKIP(1) EDIT((15)'*')(X(10),A);
        PUT SKIP(1) EDIT('* TABLE 4 - ',I,' *')(X(10),A,A,A);
        PUT SKIP(1) EDIT((15)'*')(X(10),A);
        PUT SKIP(1) EDIT('DEATH CAUSE', 'TOTAL', 'MALE', 'FEMALE')
            (X(10),A,X(30),A,X(31),A,X(29),A);
        PUT SKIP(1) EDIT('
            TT', '
            A', '
            B', '
            C',
            '
            TT', '
            A', '
            B', '
            C',
            '
            TT', '
            A', '
            B', '
            C')
            (X(21),3(X(3),4 A));
        PUT SKIP(1) EDIT((116)'*')(X(10),A);
        DO J = 1 TO 18;
            CALL TCAUSE;
            PUT SKIP(3) EDIT(BUN,((AR1(I,J,K,L) DO L = 1 TO 4)
                DO K = 1 TO 3)
                (X(10),A,3(X(3),4 F(B))));
            PUT SKIP(3);
        END;
    END;
    DO H = 1 TO 2;
        PUT PAGE;
        PUT SKIP(5) EDIT((42)'*')(X(10),A);
        PUT SKIP EDIT('* 10 AMOUNT OF DEATH BY SEXES AGE CAUSE *')
            (X(10),A);
        PUT SKIP EDIT('* 10 AGE S INTERVAL *')
            (X(10),A);
        PUT SKIP EDIT((42)'*')(X(10),A);
        DO I = 1 TO 3;
            IF I = 1 THEN DO; PUT PAGE; PUT SKIP(5); END;
            PUT SKIP(1) EDIT ((15)'*')(X(10),A);
            PUT SKIP(1) EDIT ('* TABLE 10 - ',I,' *')(X(10),A,A,A);
            PUT SKIP(1) EDIT ((15)'*')(X(10),A);
            PUT SKIP(1) EDIT('DEATH CAUSE', 'S', 'TOT', '
                -1',
                '
                1-4', '
                5-14', '
                15-24', '
                25-34', '
                35-44',
                '
                45-54', '
                55-64', '
                65-74', '
                75-', '
                N/A')
                (X(10),A,A,12 A);
            PUT SKIP EDIT((110)'*')(X(10),A);
            DO J = 1 TO 18;
                CALL TCAUSE;
                DO K = 1 TO 3;
                    IF K = 2 | K = 3 THEN BUN = (11)' ';
                    CALL TSEX;
                    PUT SKIP(2) EDIT(BUN,MF,(AR2(H,I,J,K,L) DO L = 1 TO 12))
                        (X(10),A,A,12 F(7));
                END;
            END;
        END;
    END;

```

```

END; SVD02210
END; SVD02220
DO I = 1 TO 2; SVD02230
  PUT PAGE; SVD02240
  PUT SKIP(5) EDIT((42)'*')(X(10),A); SVD02250
  PUT SKIP(1) EDIT('* 10 AMOUNT OF DEATH BY SEXES AGE CAUSE *') SVD02260
    (X(10),A); SVD02270
  PUT SKIP(1) EDIT('* BY 5 GROUP OF AGE *') SVD02280
    (X(10),A); SVD02290
  PUT SKIP(1) EDIT((42)'*')(X(10),A); SVD02300
  PUT SKIP(1) EDIT('DEATH CAUSE', ' S', ' TOT', ' -1', SVD02310
    ' 1-14', ' 15-44', ' 45-64', ' 65+', ' N/A') SVD02320
    (X(10),A,A,7 A); SVD02330
  PUT SKIP(1) EDIT((69)'=')(X(10),A); SVD02340
DO J = 1 TO 10; SVD02350
  CALL TCAUSE; SVD02360
  DO K = 1 TO 3; SVD02370
    CALL TSEX; SVD02380
    IF K = 2 | K = 3 THEN BUN = (11)' '; SVD02390
    PUT SKIP(2) EDIT(BUN,MF,(AR3(I,J,K,L) DO L = 1 TO 7)) SVD02400
      (X(10),A,A,7 F(8)); SVD02410
  END; SVD02420
END; SVD02430
END; SVD02440
PUT PAGE; SVD02450
PUT SKIP(5) EDIT((42)'*')(X(10),A); SVD02460
PUT SKIP(1) EDIT('* 7 DISTRIBUTION OF DEDTH BY SEXES,MONTH *') SVD02470
  (X(10),A); SVD02480
PUT SKIP(1) EDIT((42)'*')(X(10),A); SVD02490
DO I = 1 TO 2; SVD02500
  IF I /=1 THEN DO; PUT PAGE; PUT SKIP(5); END; SVD02510
  PUT SKIP EDIT ((15)'*')(X(10),A); SVD02520
  PUT SKIP EDIT ('* TABLE 7 - ',I,' *')(X(10),A,A,A); SVD02530
  PUT SKIP EDIT ((15)'*')(X(10),A); SVD02540
  PUT SKIP EDIT(' ICD NUMBER ', SVD02550
    'TOTAL', SVD02560
    'JANUARY', SVD02570
    'FEBRARY', SVD02580
    'MARCH', SVD02590
    'APRIL', SVD02600
    'MAY', SVD02610
    'JUNE', SVD02620
    (A,7 A); SVD02630
  PUT SKIP(1) EDIT((133)'=')(A); SVD02640
DO J = 1 TO 10; SVD02650
  CALL TCAUSE; SVD02660
  PUT SKIP(3) EDIT(BUN,((AR4(I,J,K,L) DO L = 1 TO 3) SVD02670
    DO K = 1 TO 6),AR4(I,J,7,1),AR4(I,J,7,2)) SVD02680
    (A,X(2),20 F(6)); SVD02690
  PUT SKIP(3); SVD02700
END; SVD02710
END; SVD02720
DO I = 1 TO 2; SVD02730
  PUT PAGE; SVD02740
  PUT SKIP(5) EDIT(' SVD02750

```

```

'JULY           ', SVD02760
'AUGUST        ', SVD02770
'SEPTEMBER     ', SVD02780
'OCTOBER       ', SVD02790
'NOVEMBER      ', SVD02800
'DECEMBER      ', SVD02810
'N / A         ', SVD02820
(A,7 A); SVD02830
PUT SKIP(1) EDIT((133)'=')(A); SVD02840
DO J = 1 TO 18; SVD02850
CALL TCAUSE; SVD02860
PUT SKIP(3) EDIT(AR4(I,J,7,3),((AR4(I,J,K,L) SVD02870
DO L = 1 TO 3) DO K = 8 TO 14)) SVD02880
(22 F(6)); SVD02890
PUT SKIP(3); SVD02900
END; SVD02910
END; SVD02920
PUT PAGE; SVD02930
PUT SKIP(5) EDIT((42)'*')(X(10),A); SVD02940
PUT SKIP(1) EDIT('* 5 AMOUNT OF DEATH BY CAUSE EDUCATION *') SVD02950
(X(10),A); SVD02960
PUT SKIP(1) EDIT('* BY 5 GROUP OF AGE *') SVD02970
(X(10),A); SVD02980
PUT SKIP(1) EDIT((42)'*')(X(10),A); SVD02990
PUT SKIP(1) EDIT(' ICD NUMBER', ' S', SVD03000
' TT', ' T1', ' T2', ' T3', ' T4', ' T5', SVD03010
' DT', ' D1', ' D2', ' D3', ' D4', ' D5') SVD03020
(X(10),A,A,2(X(3),6 A)); SVD03030
PUT SKIP(1) EDIT((83)'*')(X(10),A); SVD03040
DO J = 1 TO 18; SVD03050
CALL TCAUSE; SVD03060
DO K = 1 TO 3; SVD03070
CALL TSEX; SVD03080
IF K = 2 | K = 3 THEN BUN = (11)' '; SVD03090
PUT SKIP(2) EDIT(BUN,MF,((AR5(J,K,L,M) DO M = 1 TO 6) SVD03100
DO L = 1 TO 2)) SVD03110
(X(10),A,A,X(3),6 F(6),X(3),6 F(6)); SVD03120
END; SVD03130
END; SVD03140
PUT PAGE; SVD03150
PUT SKIP(5) EDIT((42)'*')(X(10),A); SVD03160
PUT SKIP(1) EDIT('* 8 THE DISTRIBUTION OF DEATH CAUSE *') SVD03170
(X(10),A); SVD03180
PUT SKIP(1) EDIT('* BY SEXES & REGION & OCCUPATION *') SVD03190
(X(10),A); SVD03200
PUT SKIP(1) EDIT((42)'*')(X(10),A); SVD03210
PUT SKIP(1) EDIT('REGION ', 'TOTAL', 'SIBU', 'GUNBU') SVD03220
(X(10),A,3 A(35)); SVD03230
PUT SKIP(1) EDIT('DEATH CAUSE S', SVD03240
' 1', ' 2', ' 3', ' 4', ' 5', SVD03250
' 1', ' 2', ' 3', ' 4', ' 5', SVD03260
' 1', ' 2', ' 3', ' 4', ' 5') SVD03270
(X(10),A,3(X(5),5 A)); SVD03280
PUT SKIP(1) EDIT((118)'*')(X(10),A); SVD03290
DO J = 1 TO 18; SVD03300

```

```

CALL TCAUSE;
DO K = 1 TO 3;
  CALL TSEX;
  IF K = 2 | K = 3 THEN BUN = (11)' ';
  PUT SKIP(2) EDIT(BUN,MF,((AR6(J,K,L,M) DO M = 1 TO 5)
    DO L = 1 TO 3))
    (X(10),A,A,3(X(5),5 F(6)));
END;
END;
PUT PAGE;
PUT SKIP(5) EDIT((42)'*')(X(10),A);
PUT SKIP(4) EDIT('* 8 THE DISTRIBUTION OF DEATH CAUSE *')
  (X(10),A);
PUT SKIP(4) EDIT('* BY SEXES & MARITAL STATUS *')
  (X(10),A);
PUT SKIP(1) EDIT((42)'*')(X(10),A);
PUT SKIP(1) EDIT((15)'*')(X(10),A);
PUT SKIP(4) EDIT('* TABLE 8 - ',I,' *')(X(10),A,A,A);
PUT SKIP(1) EDIT((15)'*')(X(10),A);
PUT SKIP EDIT('MARITAL STATUS ', ' TOTAL', ' BY DOCTOR')
  (X(10),A,2 A(40));
PUT SKIP EDIT('DEATH CAUSE S',
  ' TT', ' S', ' M', ' D', ' W', ' N/A',
  ' TT', ' S', ' M', ' D', ' W', ' N/A')
  (X(10),A,2(X(4),6 A));
PUT SKIP EDIT((93)'=')(X(10),A);
DO J = 1 TO 10;
  CALL TCAUSE;
  DO K = 1 TO 3;
    IF K = 2 | K = 3 THEN BUN = (11)' ';
    CALL TSEX;
    PUT SKIP(2) EDIT(BUN,MF,(AR7(1,J,K,1,M) DO M = 1 TO 6),
      (AR7(2,J,K,2,M) DO M = 1 TO 6))
      (X(10),A,A,X(4),6 F(6),X(4),6 F(6));
  END;
END;
TSEX: PROC;
  IF K = 1 THEN MF = ' T';
  ELSE IF K = 2 THEN MF = ' M';
  ELSE IF K = 3 THEN MF = ' F';
END TSEX;
TCAUSE: PROC;
  IF J = 1 THEN BUN = ' 001 - 999 ';
  ELSE IF J = 2 THEN BUN = ' 001 - 139 ';
  ELSE IF J = 3 THEN BUN = ' 140 - 239 ';
  ELSE IF J = 4 THEN BUN = ' 240 - 279 ';
  ELSE IF J = 5 THEN BUN = ' 280 - 289 ';
  ELSE IF J = 6 THEN BUN = ' 290 - 319 ';
  ELSE IF J = 7 THEN BUN = ' 320 - 389 ';
  ELSE IF J = 8 THEN BUN = ' 390 - 459 ';
  ELSE IF J = 9 THEN BUN = ' 460 - 519 ';
  ELSE IF J =10 THEN BUN = ' 520 - 579 ';
  ELSE IF J =11 THEN BUN = ' 580 - 629 ';
  ELSE IF J =12 THEN BUN = ' 630 - 679 ';
  ELSE IF J =13 THEN BUN = ' 680 - 709 ';

```

FILE: SVDWT714 TAB17 A1 VM/SP RELEASE 2.1 EXPRESS PUT8302+ SLU203

```
ELSE IF J =14 THEN BUN = ' 710 - 739 ' ; SVD03860
ELSE IF J =15 THEN BUN = ' 740 - 759 ' ; SVD03870
ELSE IF J =16 THEN BUN = ' 760 - 779 ' ; SVD03880
ELSE IF J =17 THEN BUN = ' 780 - 799 ' ; SVD03890
ELSE IF J =18 THEN BUN = ' 800 - 999 ' ; SVD03900
END TCAUSE; SVD03910
LAST: SVD03920
PUT PAGE; SVD03930
PUT SKIP EDIT('INPUT COUNT =',TCNT)(A,A); SVD03940
PUT SKIP EDIT('PYOBON COUNT =',OCNT)(A,A); SVD03950
PUT SKIP EDIT('MYUNGBUCOUNT =',ICNT)(A,A); SVD03960
CLOSE FILE(TF); SVD03970
END GOLD; SVD03980
/* SVD03990
//GO.TF DD UNIT=TAPE,DSN=HH,VOL=SER=123321,LABEL=(2,BLP), SVD04000
// DCB=(RECFM=FB,BLKSIZE=3200,LRECL=80),DISP=(OLD,KEEP) SVD04010
//GO.SYSPRINT DD SYSOUT=A,COPIES=2 SVD04020
//GO.SYSOUT DD SYSOUT=A SVD04030
// SVD04040
```

```

//SVDMS514 JOB TYPRUN=HOLD
//* SUMMARY BY ICD(50) ***
//SUM50 EXEC COBUCLG
//COB.SYSIN DD *
  ID DIVISION.
  PROGRAM-ID. SUMMARY.
  ENVIRONMENT DIVISION.
  INPUT-OUTPUT SECTION.
  FILE-CONTROL.
    SELECT T-F ASSIGN UT-S-TAPEIN.
    SELECT TA-F ASSIGN UT-S-TAPEOUT.
  DATA DIVISION.
  FILE SECTION.
  FD T-F BLOCK CONTAINS 40 RECORDS
    LABEL RECORD STANDARD.
01 T-R.
  02 DBIGYO.
  03 F PIC X(3).
  03 WOL PIC 99.
  03 F PIC X(11).
  03 F PIC X(4).
  03 DO PIC 99.
  03 GUN PIC 99.
  03 SEX PIC 9.
  03 F PIC X(15).
  03 OP PIC 99.
  03 CU PIC 9.
  03 F PIC XX.
  03 WON.
    05 W0 PIC 999.
      88 2T VALUES 001 THRU 139.
      88 3T VALUES 001 THRU 009.
      88 4T VALUES 010 THRU 018.
      88 5T VALUE 033.
      88 6T VALUE 036.
      88 7T VALUE 037.
      88 8T VALUE 038.
      88 9T VALUE 050.
      88 10T VALUE 055.
      88 11T VALUE 084.
      88 12T VALUES 140 THRU 208.
      88 13T VALUE 151.
      88 14T VALUE 153.
      88 15T VALUE 154.
      88 16T VALUE 162.
      88 17T VALUE 174.
      88 18T VALUE 180.
      88 19T VALUES 204 THRU 208.
      88 20T VALUE 250.
      88 21T VALUE 261.
      88 22T VALUES 262 THRU 263.
      88 23T VALUES 280 THRU 285.
      88 24T VALUES 320 THRU 322.
      88 25T VALUES 390 THRU 459.
      88 26T VALUES 390 THRU 392.
SVD00010
SVD00020
SVD00030
SVD00040
SVD00050
SVD00060
SVD00070
SVD00080
SVD00090
SVD00100
SVD00110
SVD00120
SVD00130
SVD00140
SVD00150
SVD00160
SVD00170
SVD00180
SVD00190
SVD00200
SVD00210
SVD00220
SVD00230
SVD00240
SVD00250
SVD00260
SVD00270
SVD00280
SVD00290
SVD00300
SVD00310
SVD00320
SVD00330
SVD00340
SVD00350
SVD00360
SVD00370
SVD00380
SVD00390
SVD00400
SVD00410
SVD00420
SVD00430
SVD00440
SVD00450
SVD00460
SVD00470
SVD00480
SVD00490
SVD00500
SVD00510
SVD00520
SVD00530
SVD00540
SVD00550

```

88 27T	VALUES 393 THRU 398.	SVD00560
88 28T	VALUES 401 THRU 405.	SVD00570
88 29T	VALUES 410 THRU 414.	SVD00580
88 30T	VALUE 410.	SVD00590
88 32T	VALUES 430 THRU 438.	SVD00600
88 33T	VALUE 436.	SVD00610
88 35T	VALUE 440.	SVD00620
88 36T	VALUES 480 THRU 486.	SVD00630
88 37T	VALUE 487.	SVD00640
88 38T	VALUES 490 THRU 493.	SVD00650
88 39T	VALUES 531 THRU 533.	SVD00660
88 40T	VALUES 540 THRU 543.	SVD00670
88 41T	VALUE 571.	SVD00680
88 42T	VALUES 580 THRU 589.	SVD00690
88 43T	VALUE 600.	SVD00700
88 44T	VALUES 630 THRU 639.	SVD00710
88 45T	VALUES 640 THRU 646	SVD00720
	651 THRU 676.	SVD00730
88 46T	VALUES 740 THRU 759.	SVD00740
88 47T	VALUES 760 THRU 779.	SVD00750
88 48T	VALUE 767.	SVD00760
88 49T	VALUES 780 THRU 799.	SVD00770
88 50T	VALUE 797.	SVD00780
88 51T	VALUE 799.	SVD00790
88 52T	VALUES 800 THRU 999.	SVD00800
88 53T	VALUES 800 THRU 829.	SVD00810
88 54T	VALUES 850 THRU 869	SVD00820
	950 THRU 957.	SVD00830
88 55T	VALUES 940 THRU 949.	SVD00840
88 56T	VALUES 960 THRU 989.	SVD00850
05 F	PIC XX.	SVD00860
05 BU	PIC X(3).	SVD00870
88 57T	VALUE 800 THRU 949.	SVD00880
88 58T	VALUE 810 THRU 819.	SVD00890
88 59T	VALUE 880 THRU 888.	SVD00900
88 60T	VALUE 950 THRU 959.	SVD00910
88 61T	VALUE 960 THRU 969.	SVD00920
05 F	PIC X.	SVD00930
03 W0NN	REDEFINES W0N.	SVD00940
05 WSE	PIC X(4).	SVD00950
88 31T	VALUE 4275.	SVD00960
88 34T	VALUE 4378.	SVD00970
05 F	PIC X(5).	SVD00980
03 YEA	PIC 999.	SVD00990
03 WS	PIC 99.	SVD01000
03 ISS	PIC 99.	SVD01010
02 F	PIC X(19).	SVD01020
FD	TA-F BLOCK CONTAINS 40 RECORDS	SVD01030
	LABEL RECORD STANDARD.	SVD01040
01	TA-R.	SVD01050
02	TAAA.	SVD01060
03 TTT	PIC X.	SVD01070
03 YEAR	PIC XX.	SVD01080
03 F	PIC X(30).	SVD01090
03 YR	PIC XX.	SVD01100



03 TT1	PIC X(26).	SVD01110
02 TBBB.		SVD01120
03 AA	PIC 999.	SVD01130
03 BB	PIC 999.	SVD01140
03 CC	PIC 999.	SVD01150
03 EAA	PIC 999.	SVD01160
03 EBB	PIC 999.	SVD01170
03 F	PIC X(4).	SVD01180
WORKING-STORAGE SECTION.		
77 ECNT	PIC 9(6) VALUE 0.	SVD01190
77 OCNT	PIC 9(6) VALUE 0.	SVD01200
77 YR80	PIC 9(6) VALUE 0.	SVD01210
77 YR81	PIC 9(6) VALUE 0.	SVD01220
77 YR88	PIC 9(6) VALUE 0.	SVD01230
77 YR88	PIC 9(6) VALUE 0.	SVD01240
PROCEDURE DIVISION.		
OPEN INPUT T-F.		SVD01250
OPEN OUTPUT TA-F.		SVD01260
MOVE SPACE TO TA-R.		SVD01270
READ-RTN.		SVD01280
READ T-F AT END GO TO END-RTN.		SVD01290
ADD 1 TO ECNT.		SVD01300
EXAMINE WON REPLACING ALL SPACE BY 0.		SVD01310
IF 2T	MOVE 2 TO AA.	SVD01320
IF 3T	MOVE 3 TO BB.	SVD01330
IF 4T	MOVE 4 TO BB.	SVD01340
IF 5T	MOVE 5 TO BB.	SVD01350
IF 6T	MOVE 6 TO BB.	SVD01360
IF 7T	MOVE 7 TO BB.	SVD01370
IF 8T	MOVE 8 TO BB.	SVD01380
IF 9T	MOVE 9 TO BB.	SVD01390
IF 10T	MOVE 10 TO BB.	SVD01400
IF 11T	MOVE 11 TO BB.	SVD01410
IF 12T	MOVE 12 TO AA.	SVD01420
IF 13T	MOVE 13 TO BB.	SVD01430
IF 14T	MOVE 14 TO BB.	SVD01440
IF 15T	MOVE 15 TO BB.	SVD01450
IF 16T	MOVE 16 TO BB.	SVD01460
IF 17T	MOVE 17 TO BB.	SVD01470
IF 18T	MOVE 18 TO BB.	SVD01480
IF 19T	MOVE 19 TO BB.	SVD01490
IF 20T	MOVE 20 TO AA.	SVD01500
IF 21T	MOVE 21 TO AA.	SVD01510
IF 22T	MOVE 22 TO AA.	SVD01520
IF 23T	MOVE 23 TO AA.	SVD01530
IF 24T	MOVE 24 TO AA.	SVD01540
IF 25T	MOVE 25 TO AA.	SVD01550
IF 26T	MOVE 26 TO BB.	SVD01560
IF 27T	MOVE 27 TO BB.	SVD01570
IF 28T	MOVE 28 TO BB.	SVD01580
IF 29T	MOVE 29 TO BB.	SVD01590
IF 30T	MOVE 30 TO CC.	SVD01600
IF 31T	MOVE 31 TO BB.	SVD01610
IF 32T	MOVE 32 TO BB.	SVD01620
IF 33T	MOVE 33 TO CC.	SVD01630
IF 34T	MOVE 34 TO CC.	SVD01640
		SVD01650

```

IF 35T      MOVE 35 TO BB.      SVD01660
IF 36T      MOVE 36 TO AA.      SVD01670
IF 37T      MOVE 37 TO AA.      SVD01680
IF 38T      MOVE 38 TO AA.      SVD01690
IF 39T      MOVE 39 TO AA.      SVD01700
IF 40T      MOVE 40 TO AA.      SVD01710
IF 41T      MOVE 41 TO AA.      SVD01720
IF 42T      MOVE 42 TO AA.      SVD01730
IF 43T      MOVE 43 TO AA.      SVD01740
IF 44T      MOVE 44 TO AA.      SVD01750
IF 45T      MOVE 45 TO AA.      SVD01760
IF 46T      MOVE 46 TO AA.      SVD01770
IF 47T      MOVE 47 TO AA.      SVD01780
IF 48T      MOVE 48 TO BB.      SVD01790
IF 49T      MOVE 49 TO AA.      SVD01800
IF 50T      MOVE 50 TO BB.      SVD01810
IF 51T      MOVE 51 TO BB.      SVD01820
IF 52T      MOVE 52 TO AA.      SVD01830
IF 53T      MOVE 53 TO BB.      SVD01840
IF 54T      MOVE 54 TO BB.      SVD01850
IF 55T      MOVE 55 TO BB.      SVD01860
IF 56T      MOVE 56 TO BB.      SVD01870
IF 57T      MOVE 57 TO EAA.     SVD01880
IF 58T      MOVE 58 TO EBB.     SVD01890
IF 59T      MOVE 59 TO EBB.     SVD01900
IF 60T      MOVE 60 TO EAA.     SVD01910
IF 61T      MOVE 61 TO EAA.     SVD01920
MOVE DBIGYO TO TAAA.           SVD01930
WRITE TA-R.                     SVD01940
MOVE SPACE TO TA-R.            SVD01950
ADD 1 TO OCNT.                  SVD01960
GO TO READ-RTN.                 SVD01970
END-RTN.                         SVD01980
DISPLAY " ECNT = " ECNT.         SVD01990
DISPLAY " OCNT = " OCNT.         SVD02000
CLOSE T-F TA-F.                 SVD02010
OPEN INPUT TA-F.                SVD02020
RD-RTN.                          SVD02030
READ TA-F AT END GO TO PER-RTN. SVD02040
IF YEAR = 81 IF YR = 81 ADD 1 TO YR88. SVD02050
IF YR = 81 ADD 1 TO YR80.        SVD02060
IF YR = 82 ADD 1 TO YR81.        SVD02070
GO TO RD-RTN.                   SVD02080
PER-RTN.                          SVD02090
DISPLAY " 81 YEAR " YR80.        SVD02100
DISPLAY " 82 YEAR " YR81.        SVD02110
DISPLAY " 81 82 YR " YR88.       SVD02120
CLOSE TA-F.                      SVD02130
STOP RUN.                        SVD02140
/*                                SVD02150
//GO.TAPEIN DD DSN=SVBWED,UNIT=TAPE,DISP=OLD,VOL=SER=SNOW14, SVD02160
// DCB=EROPT=SKP,LABEL=(2,BLP) SVD02170
//GO.TAPEOUT DD DSN=SNOW11,UNIT=TAPE,DISP=(NEW,KEEP), SVD02180
// VOL=(, , 99,SER=SCRATCH) SVD02190
//SYSOUT DD SYSOUT=A SVD02200

```

FILE: SVDMS514 50SUM A1 VM/SP RELEASE 2.1 EXPRESS PUT8302+ SLU203

//

SVD02210

```

//SVDWT514 JOB CLASS=A, TYPRUN=HOLD
//TAB50 EXEC PLIFCLG
//PLI.SYSIN DD *
* PROCESS GS,NEST,OPT(2) ;
/* 4. DISTRIBUTION OF DEATH BY DEATH PLACE(2) */ SVD00010
/* 8. AMOUNT OF DEATH BY SEXES & AGE & DEATH CAUSE & 10S(2) */ SVD00020
/* 8. AMOUNT OF DEATH BY SEXES & AGE & DEATH CAUSE & 5S(3) */ SVD00030
/* 7. DISTRIBUTION OF DEATH BY SEXES & MONTH (2) */ SVD00040
/* 8. DISTRIBUTION OF DEATH BY SEXES & REGION & OCCUPATION(1) */ SVD00050
/* 6. DISTRIBUTION OF DEATH BY SEXES & MARITAL STATUS(2) */ SVD00060
GOLD: PROC OPTIONS(MAIN) ; SVD00070
DCL TF FILE RECORD INPUT; SVD00080
DCL I TR, SVD00090
2 DKEY CHAR(16), SVD00100
2 DBDO CHAR(2), SVD00110
2 DBGUN CHAR(2), SVD00120
2 DJDO CHAR(2), SVD00130
2 DJGUN CHAR(2), SVD00140
2 DSEX PIC '(1)9', SVD00150
2 DBYR PIC '(3)9', SVD00160
2 DBMN PIC '(2)9', SVD00170
2 DBDYY PIC '(2)9', SVD00180
2 DDYR PIC '(3)9', SVD00190
2 DDMN PIC '(2)9', SVD00200
2 DDDYY PIC '(2)9', SVD00210
2 DPLS PIC '(1)9', SVD00220
2 DOPT PIC '(2)9', SVD00230
2 DJD PIC '(1)9', SVD00240
2 DMST PIC '(1)9', SVD00250
2 DEDT PIC '(1)9', SVD00260
2 DEACA CHAR(9), SVD00270
2 AGY PIC '(3)9', SVD00280
2 AGW PIC '(2)9', SVD00290
2 AGD PIC '(2)9', SVD00300
2 B(5) PIC '(3)9', SVD00310
2 F1 CHAR(4), SVD00320
DCL AR1(2,61,3,4) PIC '(6)9'; SVD00330
DCL AR2(2,61,3,12) PIC '(6)9'; SVD00340
DCL AR3(5,61,3,21) PIC '(6)9'; SVD00350
DCL AR4(2,61,14,3) PIC '(6)9'; SVD00360
DCL AR5(61,3,3,5) PIC '(6)9'; SVD00370
DCL AR6(2,61,3,2,6) PIC '(6)9'; SVD00380
DCL H PIC '(2)9' INIT(0); SVD00390
DCL I PIC '(1)9' INIT(0); SVD00400
DCL J PIC '(2)9' INIT(0); SVD00410
DCL K PIC '(2)9' INIT(0); SVD00420
DCL L PIC '(2)9' INIT(0); SVD00430
DCL M PIC '(2)9' INIT(0); SVD00440
DCL N PIC '(2)9' INIT(0); SVD00450
DCL O PIC '(2)9' INIT(0); SVD00460
DCL MOT(61) CHAR(11) INIT('*001 - 999 ', SVD00470
                                     '*001 - 139 ', SVD00480
                                     ' 001 - 009 ', SVD00490
                                     ' 010 - 018 ', SVD00500
                                     ' 033 ', SVD00510

```

' 036	'	SVD00560
' 037	'	SVD00570
' 038	'	SVD00580
' 050	'	SVD00590
' 055	'	SVD00600
' 084	'	SVD00610
' *140 - 208	'	SVD00620
' 151	'	SVD00630
' 153	'	SVD00640
' 154	'	SVD00650
' 162	'	SVD00660
' 174	'	SVD00670
' 180	'	SVD00680
' 204 - 208	'	SVD00690
' 250	'	SVD00700
' 261	'	SVD00710
' 262,263	'	SVD00720
' 280 - 285	'	SVD00730
' 320 - 322	'	SVD00740
' *390 - 459	'	SVD00750
' 390 392	'	SVD00760
' 393 - 398	'	SVD00770
' 401 - 405	'	SVD00780
' 410 - 414	'	SVD00790
' 410	'	SVD00800
' 427.5	'	SVD00810
' 430 - 438	'	SVD00820
' 436	'	SVD00830
' 437.8	'	SVD00840
' 440	'	SVD00850
' 480 - 486	'	SVD00860
' 487	'	SVD00870
' 490 - 493	'	SVD00880
' 531 - 533	'	SVD00890
' 540 - 543	'	SVD00900
' 571	'	SVD00910
' 580 - 589	'	SVD00920
' 600	'	SVD00930
' 630 - 639	'	SVD00940
' 640 - 646	'	SVD00950
' 740 - 759	'	SVD00960
' 760 - 779	'	SVD00970
' 767	'	SVD00980
' 780 - 799	'	SVD00990
' 797	'	SVD01000
' 799	'	SVD01010
' *800 - 999	'	SVD01020
' 800 - 829	'	SVD01030
' 850 - 869	'	SVD01040
' 940 - 949	'	SVD01050
' 960 - 989	'	SVD01060
' E800 -E949	'	SVD01070
' E810 -E819	'	SVD01080
' E880 -E889	'	SVD01090
' E950 -E959	'	SVD01100

```

                                'E960 -E969 ' ) ;
DCL MF          CHAR(2);
DCL BUN         CHAR(11);
DCL HJ          CHAR(6);
DCL TCNT        PIC '(6)9' INIT(0);
DCL OCNT        PIC '(6)9' INIT(0);
DCL ICNT        PIC '(6)9' INIT(0);
OPEN FILE(TF),FILE(SYSPRINT) LINESIZE(133) PAGESIZE(60);
ON ENDFILE(TF) GO TO GROSS;
ON ENDPAGE(SYSPRINT) BEGIN; PUT PAGE      ; PUT SKIP(5); END;
AR1 = 0;
AR2 = 0;
AR3 = 0;
AR4 = 0;
AR5 = 0;
AR6 = 0;
RD:  READ FILE(TF) INTO(TR);
      TCNT = TCNT + 1 ;
      IF SUBSTR(DEACA,1,3) = '000' THEN GOTO RD;
NN:  DO N = 1 TO 5;
      IF B(N) ^= 0 THEN
        CALL DEATH;
      END;
      CALL TOTAL;
      GOTO RD;
DEATH: PROC;
/* 4. DISTRIBUTION OF DEATH BY DEATH PLACE(2) */
      IF DJD = '1' THEN
        AR1(2,B(N),DSEX+1,DPLS+1)=AR1(2,B(N),DSEX+1,DPLS+1)+1;
        AR1(1,B(N),DSEX+1,DPLS+1) = AR1(1,B(N),DSEX+1,DPLS+1)+1;
/* 8. AMOUNT OF DEATH BY SEXES & AGE & DEATH CAUSE & 10S(2) */
        IF AGY < 1 THEN L = 2;
        ELSE IF AGY < 5 THEN L = 3;
        ELSE IF AGY < 15 THEN L = 4;
        ELSE IF AGY < 25 THEN L = 5;
        ELSE IF AGY < 35 THEN L = 6;
        ELSE IF AGY < 45 THEN L = 7;
        ELSE IF AGY < 55 THEN L = 8;
        ELSE IF AGY < 65 THEN L = 9;
        ELSE IF AGY < 75 THEN L = 10;
        ELSE L = 11;
        IF DJD = '1' THEN
          AR2(1,B(N),DSEX+1,L) = AR2(1,B(N),DSEX+1,L) + 1;
          AR2(2,B(N),DSEX+1,L) = AR2(2,B(N),DSEX+1,L) + 1;
/* 8. AMOUNT OF DEATH BY SEXES & AGE & DEATH CAUSE & 5S(3) */
        IF DJDO < 31 | DJGUN < 31 THEN I = 1;
        ELSE I = 2;
        IF AGY < 1 THEN M = 2;
        ELSE IF AGY < 5 THEN M = 3;
        ELSE IF AGY < 10 THEN M = 4;
        ELSE IF AGY < 15 THEN M = 5;
        ELSE IF AGY < 20 THEN M = 6;
        ELSE IF AGY < 25 THEN M = 7;
        ELSE IF AGY < 30 THEN M = 8;
        ELSE IF AGY < 35 THEN M = 9;

```

```

ELSE IF AGY < 40 THEN M = 10; SVD01660
ELSE IF AGY < 45 THEN M = 11; SVD01670
ELSE IF AGY < 50 THEN M = 12; SVD01680
ELSE IF AGY < 55 THEN M = 13; SVD01690
ELSE IF AGY < 60 THEN M = 14; SVD01700
ELSE IF AGY < 65 THEN M = 15; SVD01710
ELSE IF AGY < 70 THEN M = 16; SVD01720
ELSE IF AGY < 75 THEN M = 17; SVD01730
ELSE IF AGY < 80 THEN M = 18; SVD01740
ELSE IF AGY < 85 THEN M = 19; SVD01750
ELSE M = 20; SVD01760
AR3(I,B(N),DSEX+1,M) = AR3(I,B(N),DSEX+1,M) + 1; SVD01770
IF I = 1 & (DJDO = '11'|DJDO = '21'|DJDO = '22'|DJDO = '23') SVD01780
THEN SVD01790
AR3(3,B(N),DSEX+1,M) = AR3(3,B(N),DSEX+1,M) + 1; SVD01800
IF DJD = '1' THEN SVD01810
AR3(4,B(N),DSEX+1,M) = AR3(4,B(N),DSEX+1,M) + 1; SVD01820
AR3(5,B(N),DSEX+1,M) = AR3(5,B(N),DSEX+1,M) + 1; SVD01830
/* 7. DISTRIBUTION OF DEATH BY SEXES & MONTH (2) */ SVD01840
IF DJD = '1' THEN SVD01850
AR4(2,B(N),DDMN+1,DSEX+1) = AR4(2,B(N),DDMN+1,DSEX+1) + 1; SVD01860
AR4(1,B(N),DDMN+1,DSEX+1) = AR4(1,B(N),DDMN+1,DSEX+1) + 1; SVD01870
/* 8. DISTRIBUTION OF DEATH BY SEXES & REGION & OCCUPATION(1) */ SVD01880
IF DJDO < 31 | DJGUN < 31 THEN K = 2; SVD01890
ELSE K = 3; SVD01900
IF DOPT = 01 | DOPT = 02 | DOPT = 03 THEN O=1; SVD01910
ELSE IF DOPT = 04 | DOPT = 05 THEN O=2; SVD01920
ELSE IF DOPT = 06 THEN O=3; SVD01930
ELSE IF DOPT = 07 THEN O=4; SVD01940
ELSE O=5; SVD01950
AR5(B(N),DSEX+1,K,O) = AR5(B(N),DSEX+1,K,O) + 1; SVD01960
/* 6. DISTRIBUTION OF DEATH BY SEXES & MARITAL STATUS(2) */ SVD01970
IF DJD = '1' THEN SVD01980
AR6(1,B(N),DSEX+1,2,DMST+1) = AR6(1,B(N),DSEX+1,2,DMST+1) + 1; SVD01990
AR6(1,B(N),DSEX+1,1,DMST+1) = AR6(1,B(N),DSEX+1,1,DMST+1) + 1; SVD02000
END DEATH; SVD02010
TOTAL: PROC; SVD02020
IF AGY < 1 THEN L = 2; SVD02030
ELSE IF AGY < 5 THEN L = 3; SVD02040
ELSE IF AGY < 15 THEN L = 4; SVD02050
ELSE IF AGY < 25 THEN L = 5; SVD02060
ELSE IF AGY < 35 THEN L = 6; SVD02070
ELSE IF AGY < 45 THEN L = 7; SVD02080
ELSE IF AGY < 55 THEN L = 8; SVD02090
ELSE IF AGY < 65 THEN L = 9; SVD02100
ELSE IF AGY < 75 THEN L = 10; SVD02110
ELSE L = 11; SVD02120
IF DJD = '1' THEN SVD02130
AR1(2,1,DSEX+1,DPLS+1) = AR1(2,1,DSEX+1,DPLS+1) + 1; SVD02140
AR1(1,1,DSEX+1,DPLS+1) = AR1(1,1,DSEX+1,DPLS+1) + 1; SVD02150
IF DJD = '1' THEN SVD02160
AR2(1,1,DSEX+1,L) = AR2(1,1,DSEX+1,L) + 1; SVD02170
AR2(2,1,DSEX+1,L) = AR2(2,1,DSEX+1,L) + 1; SVD02180
IF AGY < 1 THEN M = 2; SVD02190
ELSE IF AGY < 5 THEN M = 3; SVD02200

```

```

ELSE IF AGY < 10 THEN M = 4; SVD02210
ELSE IF AGY < 15 THEN M = 5; SVD02220
ELSE IF AGY < 20 THEN M = 6; SVD02230
ELSE IF AGY < 25 THEN M = 7; SVD02240
ELSE IF AGY < 30 THEN M = 8; SVD02250
ELSE IF AGY < 35 THEN M = 9; SVD02260
ELSE IF AGY < 40 THEN M = 10; SVD02270
ELSE IF AGY < 45 THEN M = 11; SVD02280
ELSE IF AGY < 50 THEN M = 12; SVD02290
ELSE IF AGY < 55 THEN M = 13; SVD02300
ELSE IF AGY < 60 THEN M = 14; SVD02310
ELSE IF AGY < 65 THEN M = 15; SVD02320
ELSE IF AGY < 70 THEN M = 16; SVD02330
ELSE IF AGY < 75 THEN M = 17; SVD02340
ELSE IF AGY < 80 THEN M = 18; SVD02350
ELSE IF AGY < 85 THEN M = 19; SVD02360
ELSE M = 20; SVD02370
IF DJDO < 31 | DJGUN < 31 THEN SVD02380
AR3(1,1,DSEX+1,M) = AR3(1,1,DSEX+1,M) + 1; SVD02390
ELSE AR3(2,1,DSEX+1,M) = AR3(2,1,DSEX+1,M) + 1; SVD02400
IF I = 1 & DJD = '1' & SVD02410
(DJDO = '11'|DJDO = '21'|DJDO = '22'|DJDO = '23') SVD02420
THEN AR3(3,1,DSEX+1,M) = AR3(3,1,DSEX+1,M) + 1; SVD02430
IF DJD='1' THEN AR3(4,1,DSEX+1,M) = AR3(4,1,DSEX+1,M) + 1; SVD02440
AR3(5,1,DSEX+1,M) = AR3(5,1,DSEX+1,M) + 1; SVD02450
IF DJD = '1' THEN SVD02460
AR4(2,1,DDMN+1,DSEX+1) = AR4(2,1,DDMN+1,DSEX+1) + 1; SVD02470
AR4(1,1,DDMN+1,DSEX+1) = AR4(1,1,DDMN+1,DSEX+1) + 1; SVD02480
AR5(1,DSEX+1,K,0) = AR5(1,DSEX+1,K,0) + 1; SVD02490
IF DJD = '1' THEN SVD02500
AR6(1,1,DSEX+1,2,DMST+1) = AR6(1,1,DSEX+1,2,DMST+1) + 1; SVD02510
AR6(1,1,DSEX+1,1,DMST+1) = AR6(1,1,DSEX+1,1,DMST+1) + 1; SVD02520
END TOTAL; SVD02530
GROSS: SVD02540
AR1(*,*,*,1) = AR1(*,*,*,2) + AR1(*,*,*,3) + AR1(*,*,*,4); SVD02550
AR1(*,*,1,*) = AR1(*,*,2,*) + AR1(*,*,3,*); SVD02560
DO L = 2 TO 12; SVD02570
AR2(*,*,*,1) = AR2(*,*,*,1) + AR2(*,*,*,L); SVD02580
END; SVD02590
AR2(*,*,1,*) = AR2(*,*,2,*) + AR2(*,*,3,*); SVD02600
DO K = 2 TO 21; SVD02610
AR3(*,*,*,1) = AR3(*,*,*,1) + AR3(*,*,*,K); SVD02620
END; SVD02630
AR3(*,*,1,*) = AR3(*,*,2,*) + AR3(*,*,3,*); SVD02640
AR4(*,*,*,1) = AR4(*,*,*,2) + AR4(*,*,*,3); SVD02650
DO K = 2 TO 14; SVD02660
AR4(*,*,1,*) = AR4(*,*,1,*) + AR4(*,*,K,*); SVD02670
END; SVD02680
AR5(*,*,1,*) = AR5(*,*,2,*) + AR5(*,*,3,*); SVD02690
AR5(*,1,*,*) = AR5(*,2,*,*) + AR5(*,3,*,*); SVD02700
AR6(*,*,*,*,1) = AR6(*,*,*,*,2)+AR6(*,*,*,*,3)+AR6(*,*,*,*,4) SVD02710
+ AR6(*,*,*,*,5)+AR6(*,*,*,*,6); SVD02720
AR6(*,*,1,*,*) = AR6(*,*,2,*,*)+AR6(*,*,3,*,*); SVD02730
PRT: SVD02740
PUT PAGE; SVD02750

```



```

PUT SKIP(5) EDIT((42)'*')(X(10),A); SVD02760
PUT SKIP(1) EDIT('* 4 DISTRIBUTION OF DEATH BY DEATH PLACE *') SVD02770
(X(10),A); SVD02780
PUT SKIP(1) EDIT((42)'*')(X(10),A); SVD02790
DO I = 1 TO 2; SVD02800
  IF I = 2 THEN DO; PUT PAGE; PUT SKIP(5); END; SVD02810
  PUT SKIP(1) EDIT((15)'*')(X(10),A); SVD02820
  PUT SKIP(1) EDIT('* TABLE 4 - ',I,' *')(X(10),A,A,A); SVD02830
  PUT SKIP(1) EDIT((15)'*')(X(10),A); SVD02840
  PUT SKIP(1) EDIT('DEATH CAUSE', 'TOTAL', 'MALE', 'FEMALE') SVD02850
(X(10),A,X(30),A,X(31),A,X(29),A); SVD02860
PUT SKIP(1) EDIT(' TT', ' A', ' B', ' C', SVD02870
' TT', ' A', ' B', ' C', SVD02880
' TT', ' A', ' B', ' C') SVD02890
(X(21),3(X(3),4 A)); SVD02900
PUT SKIP(1) EDIT((117)'=')(X(10),A); SVD02910
DO J = 1 TO 61; SVD02920
  BUN = MOT(J); SVD02930
  PUT SKIP(3) EDIT(BUN,((AR1(I,J,K,L) DO L = 1 TO 4) SVD02940
DO K = 1 TO 3)) SVD02950
(X(10),A,3(X(3),4 F(8))); SVD02960
  PUT SKIP(3); SVD02970
  END; SVD02980
END; SVD02990
PUT PAGE; SVD03000
PUT SKIP(5) EDIT((42)'*')(X(10),A); SVD03010
PUT SKIP EDIT('* 10 AMOUNT OF DEATH BY SEXES AGE CAUSE *') SVD03020
(X(10),A); SVD03030
PUT SKIP EDIT('* 10 AGE S INTERVAL *') SVD03040
(X(10),A); SVD03050
PUT SKIP EDIT((42)'*')(X(10),A); SVD03060
DO I = 1 TO 2; SVD03070
  IF I = 1 THEN DO; PUT PAGE; PUT SKIP(5); END; SVD03080
  PUT SKIP(1) EDIT ((15)'*')(X(10),A); SVD03090
  PUT SKIP(1) EDIT ('* TABLE 10 - ',I,' *')(X(10),A,A,A); SVD03100
  PUT SKIP(1) EDIT ((15)'*')(X(10),A); SVD03110
  PUT SKIP(1) EDIT('DEATH CAUSE', ' S', ' TOT', ' -1', SVD03120
' 1-4', ' 5-14', ' 15-24', ' 25-34', ' 35-44', SVD03130
' 45-54', ' 55-64', ' 65-74', ' 75-', ' N/A') SVD03140
(X(10),A,A,12 A); SVD03150
PUT SKIP EDIT((107)'=')(X(10),A); SVD03160
DO J = 1 TO 61; SVD03170
  BUN = MOT(J); SVD03180
  DO K = 1 TO 3; SVD03190
    IF K = 2 | K = 3 THEN BUN = (11) ' ' ; SVD03200
    CALL TSEX; SVD03210
    PUT SKIP(2) EDIT(BUN,MF,(AR2(I,J,K,L) DO L = 1 TO 12)) SVD03220
(X(10),A,A,14 F(7)); SVD03230
  END; SVD03240
END; SVD03250
END; SVD03260
TAB8: SVD03270
PUT PAGE; SVD03280
PUT SKIP(5) EDIT((42)'*')(X(10),A); SVD03290
PUT SKIP(1) EDIT('* 8 AMOUNT OF DEATH BY SEXES AGE CAUSE *') SVD03300

```

```

(X(10),A);
PUT SKIP(1) EDIT('* 5 AGES INTERVAL *') SVD03310
(X(10),A); SVD03320
PUT SKIP(1) EDIT((42)'*')(X(10),A); SVD03330
DO I = 1 TO 5; SVD03340
  IF I = 1 THEN DO; PUT PAGE; PUT SKIP(5);END; SVD03350
  PUT SKIP EDIT((15)'*')(X(10),A); SVD03360
  PUT SKIP EDIT('* TABLE 8 -',I,'*')(X(10),A,A,A); SVD03370
  PUT SKIP EDIT((15)'*')(X(10),A); SVD03380
  PUT SKIP EDIT('DEATH CAUSE', ' S', ' TOT', ' -1', SVD03390
    ' 1-4', ' 5-9', ' 10-14', ' 15-19', ' 20-24', SVD03400
    ' 25-29', ' 30-34', ' 35-39', ' 40-44', ' 45-49', SVD03410
    ' 50-54')(X(10),A,A,13 A); SVD03420
  PUT SKIP EDIT((117)'=')(X(10),A); SVD03430
  DO J = 1 TO 61; SVD03440
    BUN= MOT(J); SVD03450
    DO K = 1 TO 3; SVD03460
      CALL TSEX; SVD03470
      IF K = 2 | K = 3 THEN BUN = (11)' '; SVD03480
      PUT SKIP(2) EDIT(BUN,MF,(AR3(I,J,K,L) DO L = 1 TO 13)) SVD03490
        (X(10),A,A,13 F(8)); SVD03500
    END; SVD03510
  END; SVD03520
  END; SVD03530
  DO I = 1 TO 5; SVD03540
    PUT PAGE; SVD03550
    PUT SKIP(5) EDIT(' 55-59', ' 60-64', ' 65-69', ' 70-74', SVD03560
      ' 75-79', ' 80-84', ' 85-', ' N/A') SVD03570
      (X(23),8 A); SVD03580
    PUT SKIP EDIT((64)'=')(X(23),A); SVD03590
    DO J = 1 TO 61; SVD03600
      DO K = 1 TO 3; SVD03610
        PUT SKIP(2) EDIT((AR3(I,J,K,L) DO L = 14 TO 21)) SVD03620
          (X(23),8 F(8)); SVD03630
      END; SVD03640
    END; SVD03650
  END; SVD03660
  PUT PAGE; SVD03670
  PUT SKIP(5) EDIT((42)'*')(X(10),A); SVD03680
  PUT SKIP EDIT('* 7 DISTRIBUTION OF DEDTH BY SEXES,MONTH *') SVD03690
    (X(10),A); SVD03700
  PUT SKIP EDIT((42)'*')(X(10),A); SVD03710
  DO I = 1 TO 2; SVD03720
    IF I = 2 THEN DO; PUT PAGE;PUT SKIP(5); END; SVD03730
    PUT SKIP EDIT((15)'*')(X(10),A); SVD03740
    PUT SKIP EDIT('* TABLE 7 -',I,'*')(X(10),A,A,A); SVD03750
    PUT SKIP EDIT((15)'*')(X(10),A); SVD03760
    PUT SKIP EDIT(' ICD NUMBER ', SVD03770
      ' TOTAL', ' ', SVD03780
      ' JANUARY', ' ', SVD03790
      ' FEBRARY', ' ', SVD03800
      ' MARCH', ' ', SVD03810
      ' APRIL', ' ', SVD03820
      ' MAY', ' ', SVD03830
      ' JUNE', ' ') SVD03840
  END; SVD03850

```

```

(A,7 A);
PUT SKIP(1) EDIT((133)'=')(A);
DO J = 1 TO 61;
  BUN = MOT(J);
  PUT SKIP(3) EDIT(BUN,((AR4(I,J,K,L) DO L = 1 TO 3)
    DO K = 1 TO 6),AR4(I,J,7,1),AR4(I,J,7,2))
    (A,X(2),20 F(6)));
  PUT SKIP(3);
END;
DO I = 1 TO 2;
PUT PAGE;
  PUT SKIP(5) EDIT('
    'JULY
    'AUGUST
    'SEPTEMBER
    'OCTOBER
    'NOVEMBER
    'DECEMBER
    'N / A
    (A,7 A);
  PUT SKIP(1) EDIT ((133)'=')(A);
  DO J = 1 TO 61;
    BUN = MOT(J);
    PUT SKIP(3) EDIT(AR4(I,J,7,3),((AR4(I,J,K,L)
      DO L = 1 TO 3) DO K = 8 TO 14))
      (22 F(6)));
    PUT SKIP(3) ;
  END;
END;
PUT PAGE;
PUT SKIP(5) EDIT((42)'*')(X(10),A);
PUT SKIP EDIT('* 8 THE DISTRIBUTION OF DEATH CAUSE *')
  (X(10),A);
PUT SKIP EDIT('* BY SEXES & REGION & OCCUPATION *')
  (X(10),A);
PUT SKIP EDIT((42)'*')(X(10),A);
PUT SKIP EDIT('REGION ' ' TOTAL ' ' SIBU ' '
  ' ' ' ' GUNBU ' '
  (X(10),A,3 A(35));
PUT SKIP EDIT('DEATH CAUSE S',
  ' 1 ' ' 2 ' ' 3 ' ' 4 ' ' 5 '
  ' 1 ' ' 2 ' ' 3 ' ' 4 ' ' 5 '
  ' 1 ' ' 2 ' ' 3 ' ' 4 ' ' 5 ')
  (X(10),A,3(X(5),5 A));
PUT SKIP EDIT((118)'=')(X(10),A);
DO J = 1 TO 61;
  BUN = MOT(J);
  DO K = 1 TO 3;
    CALL TSEX;
    IF K = 2 | K = 3 THEN BUN = (11)' ';
    PUT SKIP(2) EDIT(BUN,MF,((AR5(J,K,L,M) DO M = 1 TO 5)
      DO L = 1 TO 3))
      (X(10),A,A,3(X(5),5 F(6)));
  END;

```

SVD03860  
 SVD03870  
 SVD03880  
 SVD03890  
 SVD03900  
 SVD03910  
 SVD03920  
 SVD03930  
 SVD03940  
 SVD03950  
 SVD03960  
 SVD03970  
 SVD03980  
 SVD03990  
 SVD04000  
 SVD04010  
 SVD04020  
 SVD04030  
 SVD04040  
 SVD04050  
 SVD04060  
 SVD04070  
 SVD04080  
 SVD04090  
 SVD04100  
 SVD04110  
 SVD04120  
 SVD04130  
 SVD04140  
 SVD04150  
 SVD04160  
 SVD04170  
 SVD04180  
 SVD04190  
 SVD04200  
 SVD04210  
 SVD04220  
 SVD04230  
 SVD04240  
 SVD04250  
 SVD04260  
 SVD04270  
 SVD04280  
 SVD04290  
 SVD04300  
 SVD04310  
 SVD04320  
 SVD04330  
 SVD04340  
 SVD04350  
 SVD04360  
 SVD04370  
 SVD04380  
 SVD04390  
 SVD04400

```

END; SVD04410
PUT PAGE; SVD04420
PUT SKIP(5) EDIT((42)'*')(X(10),A); SVD04430
PUT SKIP EDIT('* 8 THE DISTRIBUTION OF DEATH CAUSE *') SVD04440
(X(10),A); SVD04450
PUT SKIP EDIT('* BY SEXES & MARITAL STATUS *') SVD04460
(X(10),A); SVD04470
PUT SKIP EDIT((42)'*')(X(10),A); SVD04480
DO I = 1; SVD04490
IF I = 2 THEN DO; PUT PAGE; PUT SKIP(5); END; SVD04500
PUT SKIP(1) EDIT((15)'*')(X(10),A); SVD04510
PUT SKIP(1) EDIT('* TABLE 8 - ',I,' *')(X(10),A,A,A); SVD04520
PUT SKIP(1) EDIT((15)'*')(X(10),A); SVD04530
PUT SKIP EDIT('MARITAL STATUS ', ' TOTAL', ' BY DOCTOR') SVD04540
(X(10),A,2 A(40)); SVD04550
PUT SKIP EDIT('DEATH CAUSE S', SVD04560
' TT', ' S', ' M', ' D', ' W', ' N/A', SVD04570
' TT', ' S', ' M', ' D', ' W', ' N/A') SVD04580
(X(10),A,2(X(4),6 A)); SVD04590
PUT SKIP EDIT((93)'=')(X(10),A); SVD04600
DO J = 1 TO 61; SVD04610
BUN = MOT(J); SVD04620
DO K = 1 TO 3; SVD04630
IF K = 2 | K = 3 THEN BUN = (11) ' '; SVD04640
CALL TSEX; SVD04650
PUT SKIP(2) EDIT(BUN,MF,((AR6(I,J,K,L,M) DO M = 1 TO 6) SVD04660
DO L = 1 TO 2)) SVD04670
(X(10),A,A,X(4),6 F(6),X(4),6 F(6)); SVD04680
END; SVD04690
END; SVD04700
END; SVD04710
TSEX: PROC; SVD04720
IF K = 1 THEN MF = ' T'; SVD04730
ELSE IF K = 2 THEN MF = ' M'; SVD04740
ELSE IF K = 3 THEN MF = ' F'; SVD04750
END TSEX; SVD04760
LAST: SVD04770
PUT PAGE; SVD04780
PUT SKIP EDIT('INPUT COUNT =',TCNT)(A,A); SVD04790
PUT SKIP EDIT('PYOBON COUNT =',OCNT)(A,A); SVD04800
PUT SKIP EDIT('MYUNGBUCOUNT =',ICNT)(A,A); SVD04810
CLOSE FILE(TF); SVD04820
END GOLD; SVD04830
/* SVD04840
//GO.TF DD UNIT=TAPE,DSN=HH,VOL=SER=123321,LABEL=(2,BLP), SVD04850
// DCB=(RECFM=FB,BLKSIZE=3200,LRECL=80),DISP=(OLD,KEEP) SVD04860
//GO.SYSOUT DD SYSOUT=A SVD04870
//GO.SYSPRINT DD SYSOUT=A SVD04880
// SVD04890

```

```

//SVDMS114 JOB TYPRUN=HOLD
//* SUMMARY BY ICD(102) ***
//SUM102 EXEC COBUCLG
//COB.SYSIN DD *
  ID DIVISION.
  PROGRAM-ID. SUMMARY.
  ENVIRONMENT DIVISION.
  INPUT-OUTPUT SECTION.
  FILE-CONTROL.
    SELECT T-F ASSIGN UT-S-TAPEIN.
    SELECT TA-F ASSIGN UT-S-TAPEOUT.
  DATA DIVISION.
  FILE SECTION.
  FD T-F BLOCK CONTAINS 40 RECORDS
    LABEL RECORD STANDARD.
  01 T-R.
    02 DBIGYD.
    03 F PIC X(3).
    03 WOL PIC 99.
    03 F PIC X(11).
    03 F PIC X(4).
    03 DD PIC 99.
    03 GUN PIC 99.
    03 SEX PIC 9.
    03 F PIC X(15).
    03 OP PIC 99.
    03 CU PIC 9.
    03 F PIC XX.
    03 WON.
    05 WO PIC 999.
  * 102 BUN RYU *
    88 1A VALUES 001 THRU 139.
    88 3A VALUES 001 THRU 009.
    88 4A VALUES 010 THRU 012.
    88 5A VALUES 013 THRU 018.
    88 6A VALUE 033.
    88 7A VALUE 036.
    88 8A VALUE 037.
    88 9A VALUE 038.
    88 10A VALUES 021 THRU 032 034 035 039 THRU 041.
    88 11A VALUE 055.
    88 12A VALUES 045 THRU 054 056 THRU 079.
    88 13A VALUE 084.
    88 14A VALUES 080 THRU 083 085 THRU 088.
    88 15A VALUES 090 THRU 099.
    88 16A VALUES 100 THRU 139.
    88 17A VALUES 140 THRU 208.
    88 18A VALUES 140 THRU 149.
    88 19A VALUE 150.
    88 20A VALUE 151.
    88 21A VALUE 153.
    88 22A VALUE 154.
    88 24A VALUE 161.
    88 25A VALUE 162.
    88 26A VALUE 174.

```

SVD00010  
 SVD00020  
 SVD00030  
 SVD00040  
 SVD00050  
 SVD00060  
 SVD00070  
 SVD00080  
 SVD00090  
 SVD00100  
 SVD00110  
 SVD00120  
 SVD00130  
 SVD00140  
 SVD00150  
 SVD00160  
 SVD00170  
 SVD00180  
 SVD00190  
 SVD00200  
 SVD00210  
 SVD00220  
 SVD00230  
 SVD00240  
 SVD00250  
 SVD00260  
 SVD00270  
 SVD00280  
 SVD00290  
 SVD00300  
 SVD00310  
 SVD00320  
 SVD00330  
 SVD00340  
 SVD00350  
 SVD00360  
 SVD00370  
 SVD00380  
 SVD00390  
 SVD00400  
 SVD00410  
 SVD00420  
 SVD00430  
 SVD00440  
 SVD00450  
 SVD00460  
 SVD00470  
 SVD00480  
 SVD00490  
 SVD00500  
 SVD00510  
 SVD00520  
 SVD00530  
 SVD00540  
 SVD00550

88 27A VALUE 180.	SVD00560
88 28A VALUES 179 182.	SVD00570
88 29A VALUE 185.	SVD00580
88 30A VALUE 188.	SVD00590
88 31A VALUES 140 THRU 199.	SVD00600
88 32A VALUES 204 THRU 208.	SVD00610
88 33A VALUES 200 THRU 203.	SVD00620
88 34A VALUES 210 THRU 239.	SVD00630
88 35A VALUE 250.	SVD00640
88 36A VALUES 240 THRU 249 251 THRU 259 270 THRU 279.	SVD00650
88 37A VALUE 261.	SVD00660
88 38A VALUES 262 263.	SVD00670
88 39A VALUES 260 264 THRU 269.	SVD00680
88 40A VALUES 280 THRU 285.	SVD00690
88 41A VALUES 286 THRU 289.	SVD00700
88 42A VALUES 290 THRU 319.	SVD00710
88 43A VALUES 320 THRU 322.	SVD00720
88 44A VALUE 340.	SVD00730
88 45A VALUE 345.	SVD00740
88 46A VALUES 323 THRU 339 341 THRU 344 346 THRU 359 360 THRU 379 380 THRU 389.	SVD00750 SVD00760
88 47A VALUES 390 THRU 459.	SVD00770
88 48A VALUES 390 THRU 392.	SVD00780
88 49A VALUES 393 THRU 398.	SVD00790
88 50A VALUES 401 THRU 405.	SVD00800
88 51A VALUE 410.	SVD00810
88 52A VALUES 411 THRU 414.	SVD00820
88 53A VALUES 415 THRU 429.	SVD00830
88 54A VALUES 430 THRU 438.	SVD00840
88 55A VALUE 440.	SVD00850
88 56A VALUES 441 THRU 444 446 THRU 448.	SVD00860
88 57A VALUES 451 THRU 453.	SVD00870
88 58A VALUES 445 449 450 454 THRU 459.	SVD00880
88 59A VALUES 460 THRU 465.	SVD00890
88 60A VALUE 466.	SVD00900
88 61A VALUES 480 THRU 486.	SVD00910
88 62A VALUE 487.	SVD00920
88 63A VALUES 490 THRU 493.	SVD00930
88 64A VALUES 470 THRU 478 488 489 494 THRU 519.	SVD00940
88 65A VALUES 531 THRU 533.	SVD00950
88 66A VALUES 540 THRU 543.	SVD00960
88 67A VALUES 550 THRU 553 560.	SVD00970
88 68A VALUE 571.	SVD00980
88 69AA VALUES 520 THRU 529 530 534 THRU 539 544 THRU 549 554 THRU 559 561 THRU 564 565 THRU 570 572 573 574 THRU 579.	SVD00990 SVD01000 SVD01010
88 70A VALUES 580 THRU 589.	SVD01020
88 71A VALUE 590.	SVD01030
88 72A VALUE 600.	SVD01040
88 73A VALUES 591 THRU 599 601 THRU 608 610 THRU 629.	SVD01050
88 74A VALUES 630 THRU 639.	SVD01060
88 75A VALUES 640 641 666.	SVD01070
88 76AA VALUE 643.	SVD01080
88 77A VALUES 670 THRU 676.	SVD01090
88 78A VALUES 640 THRU 646 651 THRU 676.	SVD01100

88 79A	VALUES 647 648 650.	SVD01110
88 80A	VALUES 680 THRU 709.	SVD01120
88 81A	VALUES 710 THRU 739.	SVD01130
88 82AA	VALUE 741.	SVD01140
88 83A	VALUES 745 THRU 747.	SVD01150
88 84AA	VALUES 740 742 THRU 744 748 THRU 759.	SVD01160
88 85A	VALUE 767.	SVD01170
88 86A	VALUES 760 THRU 766 768 THRU 779.	SVD01180
88 87A	VALUE 797.	SVD01190
88 88A	VALUES 780 THRU 796 798 THRU 799.	SVD01200
05 F	PIC XX.	SVD01210
05 BU	PIC XXX.	SVD01220
88 89A	VALUES 800 THRU 949.	SVD01230
88 90A	VALUES 810 THRU 819.	SVD01240
88 91A	VALUES 800 THRU 809 820 THRU 848.	SVD01250
88 92A	VALUES 850 THRU 869.	SVD01260
88 93A	VALUES 880 THRU 888.	SVD01270
88 94A	VALUES 890 THRU 899.	SVD01280
88 95A	VALUE 910.	SVD01290
88 96A	VALUES 919 920.	SVD01300
88 97A	VALUE 922.	SVD01310
88 98A	VALUES 870 THRU 879 911 THRU 918 900 THRU 909 921 923 THRU 929.	SVD01320
88 99A	VALUES 930 THRU 949.	SVD01330
88 100A	VALUES 950 THRU 959.	SVD01340
88 101A	VALUES 960 THRU 969.	SVD01350
88 102A	VALUES 970 THRU 999.	SVD01360
05 F	PIC X.	SVD01370
03 WGN	REDEFINES WGN.	SVD01380
05 WOSN	PIC XXXX.	SVD01390
88 2A	VALUE 0020.	SVD01400
88 23A	VALUE 1550.	SVD01410
88 69A	VALUE 5740 THRU 5751.	SVD01420
88 76A	VALUES 6424 THRU 6429.	SVD01430
88 82A	VALUE 7423.	SVD01440
88 84A	VALUE 7423.	SVD01450
05 F	PIC X(5).	SVD01460
03 YEA	PIC 999.	SVD01470
03 WS	PIC 99.	SVD01480
03 ISS	PIC 99.	SVD01490
02 F	PIC X(19).	SVD01500
FD TA-F	BLOCK CONTAINS 40 RECORDS LABEL RECORD STANDARD.	SVD01510
01 TA-R.		SVD01520
03 TTT	PIC X(61).	SVD01530
03 AA	PIC 999.	SVD01540
03 BB	PIC 999.	SVD01550
03 CC	PIC 999.	SVD01560
03 EAA	PIC 999.	SVD01570
03 EBB	PIC 999.	SVD01580
03 F	PIC X(4).	SVD01590
WORKING-STORAGE	SECTION.	SVD01600
77 ECNT	PIC 9(6) VALUE 0.	SVD01610
77 OCNT	PIC 9(6) VALUE 0.	SVD01620
PROCEDURE	DIVISION.	SVD01630
		SVD01640
		SVD01650

OPEN INPUT T-F.					SVD01660
OPEN OUTPUT TA-F.					SVD01670
MOVE SPACE TO TA-R.					SVD01680
READ-RTN.					SVD01690
READ T-F AT END GO TO END-RTN.					SVD01700
ADD 1 TO EDNT.					SVD01710
EXAMINE NON REPLACING ALL SPACE BY 0.					SVD01720
IF 1A MOVE 1 TO BB.					SVD01730
IF 2A MOVE 2 TO AA.					SVD01740
IF 3A IF NOT 2A MOVE 3 TO AA.					SVD01750
IF 4A MOVE 4 TO AA.					SVD01760
IF 5A MOVE 5 TO AA.					SVD01770
IF 6A MOVE 6 TO AA.					SVD01780
IF 7A MOVE 7 TO AA.					SVD01790
IF 8A MOVE 8 TO AA.					SVD01800
IF 9A MOVE 9 TO AA.					SVD01810
IF 10A MOVE 10 TO AA.					SVD01820
IF 11A MOVE 11 TO AA.					SVD01830
IF 12A MOVE 12 TO AA.					SVD01840
IF 13A MOVE 13 TO AA.					SVD01850
IF 14A MOVE 14 TO AA.					SVD01860
IF 15A MOVE 15 TO AA.					SVD01870
IF 16A MOVE 16 TO AA.					SVD01880
IF 17A MOVE 17 TO BB.					SVD01890
IF 18A MOVE 18 TO AA.					SVD01900
IF 19A MOVE 19 TO AA.					SVD01910
IF 20A MOVE 20 TO AA.					SVD01920
IF 21A MOVE 21 TO AA.					SVD01930
IF 22A MOVE 22 TO AA.					SVD01940
IF 23A MOVE 23 TO AA.					SVD01950
IF 24A MOVE 24 TO AA.					SVD01960
IF 25A MOVE 25 TO AA.					SVD01970
IF 26A MOVE 26 TO AA.					SVD01980
IF 27A MOVE 27 TO AA.					SVD01990
IF 28A MOVE 28 TO AA.					SVD02000
IF 29A MOVE 29 TO AA.					SVD02010
IF 30A MOVE 30 TO AA.					SVD02020
IF 31A IF NOT 18A IF NOT 19A IF NOT 20A IF NOT 21A					SVD02030
IF NOT 22A IF NOT 23A IF NOT 24A IF NOT 25A IF NOT 26A					SVD02040
IF NOT 27A IF NOT 28A IF NOT 29A IF NOT 30A					SVD02050
		MOVE 31 TO	AA.		SVD02060
IF 32A MOVE 32 TO		AA.			SVD02070
IF 33A MOVE 33 TO		AA.			SVD02080
IF 34A MOVE 34 TO		AA.			SVD02090
IF 35A MOVE 35 TO		AA.			SVD02100
IF 36A MOVE 36 TO		AA.			SVD02110
IF 37A MOVE 37 TO		AA.			SVD02120
IF 38A MOVE 38 TO		AA.			SVD02130
IF 39A MOVE 39 TO		AA.			SVD02140
IF 40A MOVE 40 TO		AA.			SVD02150
IF 41A MOVE 41 TO		AA.			SVD02160
IF 42A MOVE 42 TO		AA.			SVD02170
IF 43A MOVE 43 TO		AA.			SVD02180
IF 44A MOVE 44 TO		AA.			SVD02190
IF 45A MOVE 45 TO		AA.			SVD02200



IF	46A	MOVE	46	TO	AA.	SVD02210				
IF	47A	MOVE	47	TO	BB.	SVD02220				
IF	48A	MOVE	48	TO	AA.	SVD02230				
IF	49A	MOVE	49	TO	AA.	SVD02240				
IF	50A	MOVE	50	TO	AA.	SVD02250				
IF	51A	MOVE	51	TO	AA.	SVD02260				
IF	52A	MOVE	52	TO	AA.	SVD02270				
IF	53A	MOVE	53	TO	AA.	SVD02280				
IF	54A	MOVE	54	TO	AA.	SVD02290				
IF	55A	MOVE	55	TO	AA.	SVD02300				
IF	56A	MOVE	56	TO	AA.	SVD02310				
IF	57A	MOVE	57	TO	AA.	SVD02320				
IF	58A	MOVE	58	TO	AA.	SVD02330				
IF	59A	MOVE	59	TO	AA.	SVD02340				
IF	60A	MOVE	60	TO	AA.	SVD02350				
IF	61A	MOVE	61	TO	AA.	SVD02360				
IF	62A	MOVE	62	TO	AA.	SVD02370				
IF	63A	MOVE	63	TO	AA.	SVD02380				
IF	64A	MOVE	64	TO	AA.	SVD02390				
IF	65A	MOVE	65	TO	AA.	SVD02400				
IF	66A	MOVE	66	TO	AA.	SVD02410				
IF	67A	MOVE	67	TO	AA.	SVD02420				
IF	68A	MOVE	68	TO	AA.	SVD02430				
IF	69AA	IF NOT 69A	MOVE	69	TO	AA.	SVD02440			
IF	70A	MOVE	70	TO	AA.	SVD02450				
IF	71A	MOVE	71	TO	AA.	SVD02460				
IF	72A	MOVE	72	TO	AA.	SVD02470				
IF	73A	MOVE	73	TO	AA.	SVD02480				
IF	74A	MOVE	74	TO	AA.	SVD02490				
IF	75A	MOVE	75	TO	AA.	SVD02500				
IF	76A	OR 76AA	MOVE	76	TO	AA.	SVD02510			
IF	77A	MOVE	77	TO	AA.	SVD02520				
IF	78A	IF NOT 75A	IF NOT 76A	IF NOT 76AA	IF NOT 77A	MOVE	78	TO	AA.	SVD02540
IF	79A	MOVE	79	TO	AA.	SVD02550				
IF	80A	MOVE	80	TO	AA.	SVD02560				
IF	81A	MOVE	81	TO	AA.	SVD02570				
IF	82A	OR 82AA	MOVE	82	TO	AA.	SVD02580			
IF	83A	MOVE	83	TO	AA.	SVD02590				
IF	84A	IF NOT 82A	MOVE	84	TO	AA.	SVD02600			
IF	85A	MOVE	85	TO	AA.	SVD02610				
IF	86A	MOVE	86	TO	AA.	SVD02620				
IF	87A	MOVE	87	TO	AA.	SVD02630				
IF	88A	MOVE	88	TO	AA.	SVD02640				
IF	89A	MOVE	89	TO	EBB.	SVD02650				
IF	90A	MOVE	90	TO	EAA.	SVD02660				
IF	91A	MOVE	91	TO	EAA.	SVD02670				
IF	92A	MOVE	92	TO	EAA.	SVD02680				
IF	93A	MOVE	93	TO	EAA.	SVD02690				
IF	94A	MOVE	94	TO	EAA.	SVD02700				
IF	95A	MOVE	95	TO	EAA.	SVD02710				
IF	96A	MOVE	96	TO	EAA.	SVD02720				
IF	97A	MOVE	97	TO	EAA.	SVD02730				
IF	98A	MOVE	98	TO	EAA.	SVD02740				
IF	99A	MOVE	99	TO	EAA.	SVD02750				

```

        IF 100A MOVE 100 TO EAA. SVD02760
        IF 101A MOVE 101 TO EAA. SVD02770
        IF 102A MOVE 102 TO EAA. SVD02780
        MOVE DBIGYO TO TTT. SVD02790
        WRITE TA-R. SVD02800
        MOVE SPACE TO TA-R. SVD02810
        ADD 1 TO DCNT. SVD02820
        GO TO READ-RTN. SVD02830
END-RTN. SVD02840
        DISPLAY " ECNT = " ECNT. SVD02850
        DISPLAY " DCNT = " DCNT. SVD02860
        CLOSE T-F TA-F. SVD02870
        OPEN INPUT TA-F. SVD02880
RD-RTN. SVD02890
        READ TA-F AT END CLOSE TA-F STOP RUN. SVD02900
        DISPLAY " " TA-R. SVD02910
PER-RTN. SVD02920
        PERFORM RD-RTN 700 TIMES. SVD02930
        CLOSE TA-F. SVD02940
        STOP RUN. SVD02950
/* SVD02960
//GO.TAPEIN DD DSN=SVEWED,UNIT=TAPE,DISP=OLD,VOL=SER=SNOW14, SVD02970
// DCB=EROPT=SKP,LABEL=(2,BLF) SVD02980
//GO.TAPEOUT DD DSN=SNOW11,UNIT=TAPE,DISP=(NEW,KEEP), SVD02990
// VOL=(,,99,SER=SCRCH) SVD03000
//SYSOUT DD SYSOUT=A SVD03010
// SVD03020

```



'140 - 208	'	SVD00560
'140 - 149	'	SVD00570
'150	'	SVD00580
'151	'	SVD00590
'153	'	SVD00600
'154	'	SVD00610
'155.0	'	SVD00620
'161	'	SVD00630
'162	'	SVD00640
'174	'	SVD00650
'180	'	SVD00660
'179,182	'	SVD00670
'185	'	SVD00680
'188	'	SVD00690
'(140 - 199)-	'	SVD00700
'204 - 208	'	SVD00710
'200 - 203	'	SVD00720
'210 - 239	'	SVD00730
'250	'	SVD00740
'(290-279,270-279)-	'	SVD00750
'261	'	SVD00760
'262,263	'	SVD00770
'260,(264 - 269)	'	SVD00780
'280 - 285	'	SVD00790
'286 - 289	'	SVD00800
'290 - 319	'	SVD00810
'320 - 322	'	SVD00820
'340	'	SVD00830
'345	'	SVD00840
'323 - 339 ....	'	SVD00850
'390 - 459	'	SVD00860
'390 - 392	'	SVD00870
'393 - 398	'	SVD00880
'401 - 405	'	SVD00890
'410	'	SVD00900
'411 - 414	'	SVD00910
'415 - 429	'	SVD00920
'430 - 438	'	SVD00930
'440	'	SVD00940
'(441-444),(446-448)	'	SVD00950
'451-453	'	SVD00960
'445,449,450,454-459	'	SVD00970
'460 - 465	'	SVD00980
'466	'	SVD00990
'480 - 186	'	SVD01000
'487	'	SVD01010
'490 - 493	'	SVD01020
'470-478,488,489,494-	'	SVD01030
'531 - 533	'	SVD01040
'540 - 543	'	SVD01050
'550-553,560	'	SVD01060
'571	'	SVD01070
'520-529 ....	'	SVD01080
'580 - 589	'	SVD01090
'590	'	SVD01100

```

        '600', SVD01110
        '591-599,601-608,610-', SVD01120
        '630 - 639', SVD01130
        '640,641,666', SVD01140
        '642.4 - 642.9 ,643', SVD01150
        '670 - 676', SVD01160
        '(640-646,651-676)-', SVD01170
        '647,648,650', SVD01180
        '680 - 709', SVD01190
        '710 - 739', SVD01200
        '741,742.3', SVD01210
        '745 - 747', SVD01220
        '(740-759)-', SVD01230
        '767', SVD01240
        '(760-779)- 767', SVD01250
        '797', SVD01260
        '(780-799)- 797', SVD01270
        'E800-E949', SVD01280
        'E810-E819', SVD01290
        '(E800-848)-(810-E819)', SVD01300
        'E850-E869', SVD01310
        'E880-E888', SVD01320
        'E890-E899', SVD01330
        'E910', SVD01340
        'E919,E920', SVD01350
        'E922', SVD01360
        'E870-E879', SVD01370
        'E930-E949', SVD01380
        'E950-E959', SVD01390
        'E960-E969', SVD01400
        'E970-E999', SVD01410
DCL MF CHAR(2); SVD01420
DCL BUN CHAR(20); SVD01430
DCL TCNT PIC '(6)9' INIT(0); SVD01440
DCL OCNT PIC '(6)9' INIT(0); SVD01450
DCL ICNT PIC '(6)9' INIT(0); SVD01460
OPEN FILE(TF),FILE(SYSPRINT) LINESIZE(133) PAGESIZE(60); SVD01470
ON ENDFILE(TF) GO TO GROSS; SVD01480
ON ENDPAGE(SYSPRINT) BEGIN; PUT PAGE; PUT SKIP(5); END; SVD01490
AR = 0; SVD01500
RD: READ FILE(TF) INTO(TR); SVD01510
    TCNT = TCNT + 1 ; SVD01520
    IF SUBSTR(DEACA,1,3) = '000' THEN GOTO RD; SVD01530
    IF DJD ^= '1' THEN GOTO RD; SVD01540
NN: DO N = 1 TO 5; SVD01550
    IF B(N) ^= 0 THEN SVD01560
        CALL DEATH; SVD01570
    END; SVD01580
    IF AGY < 4 THEN L = 2; SVD01590
    ELSE IF AGY < 5 THEN L = 3; SVD01600
    ELSE IF AGY < 15 THEN L = 4; SVD01610
    ELSE IF AGY < 25 THEN L = 5; SVD01620
    ELSE IF AGY < 35 THEN L = 6; SVD01630
    ELSE IF AGY < 45 THEN L = 7; SVD01640
    ELSE IF AGY < 55 THEN L = 8; SVD01650

```

```

ELSE IF AGY < 65 THEN L = 9;
ELSE IF AGY < 75 THEN L = 10;
ELSE L = 11;
AR(I,DSEX+1,L) = AR(I,DSEX+1,L) + 1;
GOTO RD;
DEATH: PROC;
/*
/* ADDITION BY NUMBER OF I.C.D. (102)
/*
IF AGY < 1 THEN L = 2;
ELSE IF AGY < 5 THEN L = 3;
ELSE IF AGY < 15 THEN L = 4;
ELSE IF AGY < 25 THEN L = 5;
ELSE IF AGY < 35 THEN L = 6;
ELSE IF AGY < 45 THEN L = 7;
ELSE IF AGY < 55 THEN L = 8;
ELSE IF AGY < 65 THEN L = 9;
ELSE IF AGY < 75 THEN L = 10;
ELSE L = 11;
AR(B(N)+1,DSEX+1,L) = AR(B(N)+1,DSEX+1,L) + 1;
END DEATH;
GROSS:
DO L = 2 TO 12;
AR(*,*,L) = AR(*,*,L) + AR(*,*,L);
END;
AR(*,1,*) = AR(*,2,*) + AR(*,3,*);
PRT:
PUT PAGE;
PUT SKIP(5) EDIT((42)'*')(X(10),A);
PUT SKIP EDIT('* ** AMOUNT OF DEATH BY AGE(10 YEARS) *')
(X(10),A);
PUT SKIP EDIT('* BY NUMBER OF I.C.D (102) *')
(X(10),A);
PUT SKIP EDIT((42)'*')(X(10),A);
PUT SKIP(1) EDIT(' DEATH CAUSE ',
'S', ' TOT', '-1',
' 1-4', ' 5-14', ' 15-24', ' 25-34', ' 35-44',
' 45-54', ' 55-64', ' 65-74', ' 75-', ' N/A')
(X(10),A,A,12 A);
PUT SKIP EDIT((107)'=')(X(10),A);
DO J = 1 TO 103;
BUN = MOT(J);
DO K = 1 TO 3;
IF K = 2 | K = 3 THEN BUN = (11)' ';
CALL TSEX;
PUT SKIP(2) EDIT(BUN,MF,(AR(J,K,L) DO L = 1 TO 12))
(X(10),A,A,14 F(7));
END;
END;
TSEX: PROC;
IF K = 1 THEN MF = ' T';
ELSE IF K = 2 THEN MF = ' M';
ELSE IF K = 3 THEN MF = ' F';
END TSEX;
LAST:

```

SVD01660  
SVD01670  
SVD01680  
SVD01690  
SVD01700  
SVD01710  
SVD01720  
SVD01730  
SVD01740  
SVD01750  
SVD01760  
SVD01770  
SVD01780  
SVD01790  
SVD01800  
SVD01810  
SVD01820  
SVD01830  
SVD01840  
SVD01850  
SVD01860  
SVD01870  
SVD01880  
SVD01890  
SVD01900  
SVD01910  
SVD01920  
SVD01930  
SVD01940  
SVD01950  
SVD01960  
SVD01970  
SVD01980  
SVD01990  
SVD02000  
SVD02010  
SVD02020  
SVD02030  
SVD02040  
SVD02050  
SVD02060  
SVD02070  
SVD02080  
SVD02090  
SVD02100  
SVD02110  
SVD02120  
SVD02130  
SVD02140  
SVD02150  
SVD02160  
SVD02170  
SVD02180  
SVD02190  
SVD02200

FILE: SVDWT014 TAB102 A1 VM/SP RELEASE 2.1 EXPRESS PUT8302+ SLU203

```
PUT PAGE; SVD02210
PUT SKIP EDIT('INPUT COUNT =',TCNT)(A,A); SVD02220
PUT SKIP EDIT('PYOBON COUNT =',OCNT)(A,A); SVD02230
PUT SKIP EDIT('MYUNGBUCOUNT =',ICNT)(A,A); SVD02240
CLOSE FILE(TF); SVD02250
END GOLD; SVD02260
/* SVD02270
//GO.TF DD UNIT=TAFE,DSN=HH,VOL=SER=123321,LABEL=(2,BLP), SVD02280
// DCB=(RECFM=FB,BLKSIZE=3200,LRECL=80),DISP=(OLD,KEEP) SVD02290
//GO.SYSPRINT DD SYSOUT=A,COPIES=2 SVD02300
//GO.SYSOUT DD SYSOUT=A SVD02310
// SVD02320
```

```

//SVDW314 JOB TYPRUN=HOLD                                SVD00010
//*   SUMMARY BY ICD(307)                                SVD00020
//SUM307 EXEC COBUCLG                                    SVD00030
//COB.SYSIN DD *                                         SVD00040
    ID DIVISION.                                         SVD00050
    PROGRAM-ID. SUMMARY.                                  SVD00060
    ENVIRONMENT DIVISION.                                 SVD00070
    INPUT-OUTPUT SECTION.                                SVD00080
    FILE-CONTROL.                                        SVD00090
        SELECT T-F ASSIGN UT-S-TAPEIN.                   SVD00100
        SELECT TA-F ASSIGN UT-S-TAPEOUT.                  SVD00110
    DATA DIVISION.                                      SVD00120
    FILE SECTION.                                        SVD00130
    FD T-F BLOCK CONTAINS 40 RECORDS                     SVD00140
        LABEL RECORD STANDARD.                           SVD00150
01  T-R.                                                 SVD00160
    02  DBIGYO.                                          SVD00170
    03  F          PIC X(3).                             SVD00180
    03  WOL        PIC 99.                               SVD00190
    03  F          PIC X(11).                            SVD00200
    03  F          PIC X(4).                             SVD00210
    03  DO         PIC 99.                               SVD00220
    03  GUN        PIC 99.                               SVD00230
    03  SEX        PIC 9.                                SVD00240
    03  F PIC X(15).                                    SVD00250
    03  OP PIC 99.                                       SVD00260
    03  CU         PIC 9.                                SVD00270
    03  F          PIC XX.                               SVD00280
    03  WON.                                             SVD00290
        05  WO          PIC 999.                         SVD00300
            88  1T      VALUES 001 THRU 009.           SVD00310
                88  2T      VALUE 001.                 SVD00320
                88  4T      VALUE 004.                 SVD00330
                88  5T      VALUES 003 005.           SVD00340
                88  6T      VALUE 006.                 SVD00350
                88  7T      VALUES 007 008.           SVD00360
                88  8T      VALUE 009.                 SVD00370
            88  9T      VALUES 010 THRU 018.           SVD00380
                88  10T     VALUE 011.                 SVD00390
                88  11T     VALUES 010 012.           SVD00400
                88  12T     VALUE 013.                 SVD00410
                88  13T     VALUE 014.                 SVD00420
                88  14T     VALUE 015.                 SVD00430
                88  15T     VALUE 016.                 SVD00440
            88  16T     VALUES 020 THRU 041.           SVD00450
                88  17T     VALUE 020.                 SVD00460
                88  18T     VALUE 023.                 SVD00470
                88  19T     VALUE 030.                 SVD00480
                88  20T     VALUE 032.                 SVD00490
                88  21T     VALUE 033.                 SVD00500
                88  22T     VALUES 034 035.           SVD00510
                88  23T     VALUE 036.                 SVD00520
                88  24T     VALUE 037.                 SVD00530
                88  25T     VALUE 038.                 SVD00540
            88  26T     VALUES 045 THRU 079.           SVD00550

```



88 27T	VALUE	045.	SVD00560
88 28T	VALUE	050.	SVD00570
88 29T	VALUE	055.	SVD00580
88 30T	VALUE	056.	SVD00590
88 31T	VALUE	060.	SVD00600
88 32T	VALUES	062 THRU 064.	SVD00610
88 33T	VALUE	070.	SVD00620
88 34T	VALUE	071.	SVD00630
88 35T	VALUE	076.	SVD00640
88 36T	VALUES	080 THRU 088.	SVD00650
88 37T	VALUE	080.	SVD00660
88 38T	VALUES	081 THRU 083.	SVD00670
88 39T	VALUE	084.	SVD00680
88 40T	VALUE	085.	SVD00690
88 41T	VALUE	086.	SVD00700
88 42T	VALUES	090 THRU 099.	SVD00710
88 43T	VALUES	090 THRU 097.	SVD00720
88 44T	VALUE	098.	SVD00730
88 45T	VALUES	100 THRU 139.	SVD00740
88 46T	VALUES	100 THRU 104.	SVD00750
88 47T	VALUES	110 THRU 118.	SVD00760
88 48T	VALUE	120.	SVD00770
88 49T	VALUE	122.	SVD00780
88 50T	VALUE	125.	SVD00790
88 51T	VALUE	126.	SVD00800
88 52T	VALUES	121 123 124 127 THRU 129.	SVD00810
88 53T	VALUE	137.	SVD00820
88 54T	VALUE	138.	SVD00830
88 55T	VALUES	140 THRU 149.	SVD00840
88 56T	VALUES	150 THRU 159.	SVD00850
88 57T	VALUE	150.	SVD00860
88 58T	VALUE	151.	SVD00870
88 59T	VALUE	152.	SVD00880
88 60T	VALUE	153.	SVD00890
88 61T	VALUE	154.	SVD00900
88 63T	VALUE	157.	SVD00910
88 64T	VALUES	160 THRU 165.	SVD00920
88 65T	VALUE	161.	SVD00930
88 66T	VALUE	162.	SVD00940
88 67T	VALUES	170 THRU 175.	SVD00950
88 68T	VALUE	170.	SVD00960
88 69T	VALUE	172.	SVD00970
88 70T	VALUE	173.	SVD00980
88 71T	VALUE	174.	SVD00990
88 72T	VALUES	179 THRU 189.	SVD01000
88 73T	VALUE	180.	SVD01010
88 74T	VALUE	181.	SVD01020
88 75T	VALUES	179 182.	SVD01030
88 76T	VALUE	183.	SVD01040
88 77T	VALUE	185.	SVD01050
88 78T	VALUE	186.	SVD01060
88 79T	VALUE	188.	SVD01070
88 80T	VALUES	190 THRU 199.	SVD01080
88 81T	VALUE	191.	SVD01090
88 82T	VALUES	200 THRU 208.	SVD01100

88 83T	VALUE	201.	SVD01110
88 84T	VALUES	204 THRU 208.	SVD01120
88 85T	VALUES	210 THRU 229.	SVD01130
88 86T	VALUE	216.	SVD01140
88 87T	VALUE	217.	SVD01150
88 88T	VALUES	218 219.	SVD01160
88 89T	VALUE	220.	SVD01170
88 90T	VALUE	223.	SVD01180
88 91T	VALUE	225.	SVD01190
88 92T	VALUE	226.	SVD01200
88 93T	VALUES	230 THRU 234.	SVD01210
88 94T	VALUES	235 THRU 239.	SVD01220
88 95T	VALUES	240 THRU 259 270 THRU 279.	SVD01230
88 96T	VALUES	240 THRU 246.	SVD01240
88 97T	VALUE	250.	SVD01250
88 100T	VALUES	260 THRU 269.	SVD01260
88 101T	VALUE	260.	SVD01270
88 102T	VALUE	261.	SVD01280
88 103T	VALUES	262 263.	SVD01290
88 104T	VALUES	264 THRU 269.	SVD01300
88 105T	VALUES	280 THRU 289.	SVD01310
88 106T	VALUES	280 THRU 285.	SVD01320
88 107T	VALUES	290 THRU 319.	SVD01330
88 108T	VALUE	290.	SVD01340
88 109T	VALUE	295.	SVD01350
88 110T	VALUE	296.	SVD01360
88 111T	VALUES	291 THRU 294 297 THRU 299.	SVD01370 SVD01380
88 112T	VALUES	300 301.	SVD01390
88 113T	VALUE	303.	SVD01400
88 114T	VALUE	304.	SVD01410
88 115T	VALUE	306.	SVD01420
88 116T	VALUES	317 THRU 319.	SVD01430
88 117T	VALUES	320 THRU 359.	SVD01440
88 118T	VALUES	320 THRU 322.	SVD01450
88 119T	VALUE	332.	SVD01460
88 120T	VALUES	330 331 333 THRU 336.	SVD01470
88 121T	VALUE	340.	SVD01480
88 122T	VALUES	343 THRU 344.	SVD01490
88 123T	VALUE	345.	SVD01500
88 124T	VALUES	360 THRU 379.	SVD01510
88 125T	VALUE	365.	SVD01520
88 126T	VALUE	366.	SVD01530
88 127T	VALUE	369.	SVD01540
88 129T	VALUE	375.	SVD01550
88 130T	VALUE	378.	SVD01560
88 131T	VALUES	380 THRU 389.	SVD01570
88 132T	VALUES	381 THRU 383.	SVD01580
88 133T	VALUE	389.	SVD01590
88 134T	VALUES	390 THRU 398.	SVD01600
88 135T	VALUES	390 THRU 392.	SVD01610
88 136T	VALUES	393 THRU 398.	SVD01620
88 137T	VALUES	401 THRU 405.	SVD01630
88 138T	VALUES	402 404.	SVD01640
88 139T	VALUES	410 THRU 414.	SVD01650

08 140T	VALUE	410.	SVD01660
08 141T	VALUES	415 THRU 429.	SVD01670
08 143T	VALUE	427.	SVD01680
08 145T	VALUES	430 THRU 438.	SVD01690
08 146T	VALUE	430.	SVD01700
08 147T	VALUES	431 THRU 432.	SVD01710
08 148T	VALUES	433 THRU 434.	SVD01720
08 149T	VALUE	436.	SVD01730
08 152T	VALUES	440 THRU 459.	SVD01740
08 153T	VALUE	440.	SVD01750
08 154T	VALUE	444.	SVD01760
08 155T	VALUES	441 THRU 443 446 THRU 448.	SVD01770
08 156T	VALUES	451 THRU 453.	SVD01780
08 157T	VALUE	454.	SVD01790
08 158T	VALUE	455.	SVD01800
08 159T	VALUES	460 THRU 465 470 THRU 478.	SVD01810
08 160T	VALUE	463.	SVD01820
08 161T	VALUE	464.	SVD01830
08 162T	VALUES	460 THRU 462 465.	SVD01840
08 163T	VALUES	470 THRU 471.	SVD01850
08 164T	VALUES	472 THRU 473.	SVD01860
08 165T	VALUE	474.	SVD01870
08 166T	VALUES	466 480 THRU 519.	SVD01880
08 167T	VALUE	466.	SVD01890
08 168T	VALUES	480 THRU 486.	SVD01900
08 169T	VALUE	487.	SVD01910
08 170T	VALUES	490 THRU 493.	SVD01920
08 171T	VALUE	494.	SVD01930
08 172T	VALUES	495 THRU 496.	SVD01940
08 173T	VALUES	500 THRU 508.	SVD01950
08 174T	VALUE	511.	SVD01960
08 175T	VALUES	520 THRU 529.	SVD01970
08 176T	VALUES	520 THRU 525.	SVD01980
08 177T	VALUE	526.	SVD01990
08 178T	VALUES	530 THRU 579.	SVD02000
08 179T	VALUE	530.	SVD02010
08 180T	VALUES	531 THRU 533.	SVD02020
08 181T	VALUES	540 THRU 543.	SVD02030
08 182T	VALUES	550 THRU 553.	SVD02040
08 183T	VALUE	560.	SVD02050
08 184T	VALUE	562.	SVD02060
08 185T	VALUE	564.	SVD02070
08 186T	VALUE	571.	SVD02080
08 188T	VALUES	580 THRU 599.	SVD02090
08 189T	VALUES	580 THRU 589.	SVD02100
08 190T	VALUE	590.	SVD02110
08 191T	VALUES	592 594.	SVD02120
08 192T	VALUE	595.	SVD02130
08 193T	VALUES	600 THRU 608.	SVD02140
08 194T	VALUE	600.	SVD02150
08 195T	VALUE	603.	SVD02160
08 196T	VALUE	605.	SVD02170
08 197T	VALUE	606.	SVD02180
08 198T	VALUES	610 THRU 629.	SVD02190
08 199T	VALUES	610 THRU 611.	SVD02200

88 202T	VALUES 615 THRU 616.	SVD02210
88 203T	VALUE 618.	SVD02220
88 205T	VALUE 628.	SVD02230
88 206T	VALUES 630 THRU 639.	SVD02240
88 207T	VALUE 634.	SVD02250
88 208T	VALUE 635.	SVD02260
88 209T	VALUE 636.	SVD02270
88 210T	VALUES 640 THRU 646 651 THRU 676.	SVD02280
88 211T	VALUES 640 641 666.	SVD02290
88 214T	VALUE 660.	SVD02300
88 215T	VALUES 670 THRU 676.	SVD02310
88 216T	VALUES 647 THRU 648.	SVD02320
88 217T	VALUE 650.	SVD02330
88 218T	VALUES 680 THRU 709.	SVD02340
88 219T	VALUES 680 THRU 686.	SVD02350
88 220T	VALUES 710 THRU 739.	SVD02360
88 221T	VALUE 714.	SVD02370
88 222T	VALUES 710 THRU 713 715 716.	SVD02380
88 223T	VALUES 717 THRU 719.	SVD02390
88 226T	VALUES 725 THRU 729.	SVD02400
88 227T	VALUE 730.	SVD02410
88 228T	VALUES 734 THRU 736.	SVD02420
88 229T	VALUES 740 THRU 759.	SVD02430
88 232T	VALUES 745 THRU 747.	SVD02440
88 233T	VALUE 749.	SVD02450
88 234T	VALUES 750 THRU 751.	SVD02460
88 238T	VALUES 760 THRU 779.	SVD02470
88 239T	VALUE 760.	SVD02480
88 240T	VALUES 761 THRU 763.	SVD02490
88 241T	VALUES 764 THRU 765.	SVD02500
88 242T	VALUE 767.	SVD02510
88 243T	VALUES 768 THRU 770.	SVD02520
88 244T	VALUE 773.	SVD02530
88 245T	VALUES 780 THRU 799.	SVD02540
88 251T	VALUE 797.	SVD02550
88 254T	VALUES 800 THRU 829.	SVD02560
88 255T	VALUES 800 THRU 804.	SVD02570
88 256T	VALUES 805 THRU 809.	SVD02580
88 257T	VALUES 812 THRU 813.	SVD02590
88 258T	VALUE 820.	SVD02600
88 259T	VALUE 821.	SVD02610
88 260T	VALUES 823 THRU 824.	SVD02620
88 261T	VALUES 810 THRU 811 814 THRU 819 822 825 829.	SVD02630
88 262T	VALUES 830 THRU 848.	SVD02640
88 263T	VALUES 850 THRU 869 950 THRU 957.	SVD02650
88 264T	VALUE 850.	SVD02660
88 265T	VALUES 851 THRU 854 950 THRU 951.	SVD02670
88 266T	VALUES 870 THRU 904.	SVD02680
88 267T	VALUES 870 THRU 873.	SVD02690
88 268T	VALUES 880 THRU 887.	SVD02700
88 269T	VALUES 890 THRU 897.	SVD02710
88 270T	VALUES 930 THRU 939.	SVD02720
88 271T	VALUES 940 THRU 949.	SVD02730
88 272T	VALUE 940.	SVD02740
88 273T	VALUE 944.	SVD02750

88 274T	VALUES 960 THRU 989.	SVD02760
88 275T	VALUES 960 THRU 979.	SVD02770
88 276T	VALUES 996 THRU 999.	SVD02780
88 277T	VALUES 910 THRU 929.	SVD02790
	958 THRU 959 990 THRU 995.	SVD02800
88 278T	VALUES 905 THRU 909.	SVD02810
05 F PIC XX.		SVD02820
05 BU PIC XXX.		SVD02830
88 279T	VALUE 800 THRU 848.	SVD02840
88 280T	VALUE 800 THRU 807.	SVD02850
88 281T	VALUE 810 THRU 819.	SVD02860
88 282T	VALUE 824 THRU 829.	SVD02870
88 283T	VALUE 830 THRU 838.	SVD02880
88 284T	VALUE 840 THRU 845.	SVD02890
88 285T	VALUE 850 THRU 869.	SVD02900
88 286T	VALUE 850 THRU 858.	SVD02910
88 287T	VALUE 860 THRU 866.	SVD02920
88 288T	VALUE 867 THRU 869.	SVD02930
88 289T	VALUE 870 THRU 879.	SVD02940
88 290T	VALUE 880 THRU 888.	SVD02950
88 291T	VALUE 890 THRU 899.	SVD02960
88 292T	VALUE 900 THRU 929.	SVD02970
88 293T	VALUE 900 THRU 909.	SVD02980
88 294T	VALUE 910.	SVD02990
88 295T	VALUE 914 THRU 915.	SVD03000
88 296T	VALUE 919 THRU 920.	SVD03010
88 297T	VALUE 922.	SVD03020
88 298T	VALUE 930 THRU 949.	SVD03030
88 299T	VALUE 950 THRU 959.	SVD03040
88 300T	VALUE 960 THRU 969.	SVD03050
88 301T	VALUE 970 THRU 999.	SVD03060
88 302T	VALUE 980 THRU 989.	SVD03070
88 303T	VALUE 990 THRU 999.	SVD03080
05 F PIC X.		SVD03090
03 W0NN REDEFINES W0N.		SVD03100
05 W0SN PIC XXXX.		SVD03110
88 3T	VALUE 0020.	SVD03120
88 62T	VALUE 1550.	SVD03130
88 98T	VALUE 2720 THRU 2724.	SVD03140
88 99T	VALUE 2780.	SVD03150
88 128T	VALUE 3720 THRU 3723.	SVD03160
88 142T	VALUE 4151.	SVD03170
88 144T	VALUE 4275.	SVD03180
88 150T	VALUE 4370.	SVD03190
88 151T	VALUE 4378.	SVD03200
88 187T	VALUE 5740 THRU 5751.	SVD03210
88 200T	VALUE 6140 THRU 6142.	SVD03220
88 201T	VALUE 6143 THRU 6149.	SVD03230
88 204T	VALUE 6260 THRU 6265.	SVD03240
88 212T	VALUES 6424 THRU 6429 6430 THRU 6439.	SVD03250
88 213T	VALUE 6466.	SVD03260
88 224T	VALUE 7200.	SVD03270
88 225T	VALUE 7201 THRU 7249.	SVD03280
88 230T	VALUES 7423 7410 THRU 7419.	SVD03290
88 231T	VALUES 7400 THRU 7409 7420 THRU 7422	SVD03300

		7424 THRU 7429.	SVD03310
08	235T	VALUE 7525.	SVD03320
08	236T	VALUE 7543.	SVD03330
08	237T	VALUES 7540 THRU 7542 7544 THRU 7569.	SVD03340
08	246T	VALUE 7806.	SVD03350
08	247T	VALUE 7850 THRU 7853.	SVD03360
08	248T	VALUE 7880.	SVD03370
08	249T	VALUE 7882.	SVD03380
08	250T	VALUE 7890.	SVD03390
08	252T	VALUE 7980.	SVD03400
08	253T	VALUE 7991.	SVD03410
05	F	PIC X(5).	SVD03420
03	YEA	PIC 999.	SVD03430
03	WS	PIC 99.	SVD03440
03	ISS	PIC 99.	SVD03450
02	F	PIC X(19).	SVD03460
FD	TA-F BLOCK CONTAINS 40 RECORDS		SVD03470
	LABEL RECORD STANDARD.		SVD03480
01	TA-R.		SVD03490
03	TTT	PIC X(61).	SVD03500
03	AA	PIC 999.	SVD03510
03	BB	PIC 999.	SVD03520
03	CC	PIC 999.	SVD03530
03	EAA	PIC 999.	SVD03540
03	EBB	PIC 999.	SVD03550
03	F	PIC X(4).	SVD03560
	WORKING-STORAGE SECTION.		SVD03570
77	ECNT	PIC 9(6) VALUE 0.	SVD03580
77	OCNT	PIC 9(6) VALUE 0.	SVD03590
	PROCEDURE DIVISION.		SVD03600
	OPEN INPUT T-F.		SVD03610
	OPEN OUTPUT TA-F.		SVD03620
	MOVE SPACE TO TA-R.		SVD03630
	READ-RTN.		SVD03640
	READ T-F AT END GO TO END-RTN.		SVD03650
	ADD 1 TO ECNT.		SVD03660
	EXAMINE WON REPLACING ALL SPACE BY 0.		SVD03670
	IF 1T	MOVE 1 TO AA.	SVD03680
	IF 2T	MOVE 2 TO BB.	SVD03690
	IF 3T	MOVE 3 TO BB.	SVD03700
	IF 4T	MOVE 4 TO BB.	SVD03710
	IF 5T	MOVE 5 TO BB.	SVD03720
	IF 6T	MOVE 6 TO BB.	SVD03730
	IF 7T	MOVE 7 TO BB.	SVD03740
	IF 8T	MOVE 8 TO BB.	SVD03750
	IF 9T	MOVE 9 TO AA.	SVD03760
	IF 10T	MOVE 10 TO BB.	SVD03770
	IF 11T	MOVE 11 TO BB.	SVD03780
	IF 12T	MOVE 12 TO BB.	SVD03790
	IF 13T	MOVE 13 TO BB.	SVD03800
	IF 14T	MOVE 14 TO BB.	SVD03810
	IF 15T	MOVE 15 TO BB.	SVD03820
	IF 16T	MOVE 16 TO AA.	SVD03830
	IF 17T	MOVE 17 TO BB.	SVD03840
	IF 18T	MOVE 18 TO BB.	SVD03850

IF 19T	MOVE	19 TO BB.	SVD03860
IF 20T	MOVE	20 TO BB.	SVD03870
IF 21T	MOVE	21 TO BB.	SVD03880
IF 22T	MOVE	22 TO BB.	SVD03890
IF 23T	MOVE	23 TO BB.	SVD03900
IF 24T	MOVE	24 TO BB.	SVD03910
IF 25T	MOVE	25 TO BB.	SVD03920
IF 26T	MOVE	26 TO AA.	SVD03930
IF 27T	MOVE	27 TO BB.	SVD03940
IF 28T	MOVE	28 TO BB.	SVD03950
IF 29T	MOVE	29 TO BB.	SVD03960
IF 30T	MOVE	30 TO BB.	SVD03970
IF 31T	MOVE	31 TO BB.	SVD03980
IF 32T	MOVE	32 TO BB.	SVD03990
IF 33T	MOVE	33 TO BB.	SVD04000
IF 34T	MOVE	34 TO BB.	SVD04010
IF 35T	MOVE	35 TO BB.	SVD04020
IF 36T	MOVE	36 TO AA.	SVD04030
IF 37T	MOVE	37 TO BB.	SVD04040
IF 38T	MOVE	38 TO BB.	SVD04050
IF 39T	MOVE	39 TO BB.	SVD04060
IF 40T	MOVE	40 TO BB.	SVD04070
IF 41T	MOVE	41 TO BB.	SVD04080
IF 42T	MOVE	42 TO AA.	SVD04090
IF 43T	MOVE	43 TO BB.	SVD04100
IF 44T	MOVE	44 TO BB.	SVD04110
IF 45T	MOVE	45 TO AA.	SVD04120
IF 46T	MOVE	46 TO BB.	SVD04130
IF 47T	MOVE	47 TO BB.	SVD04140
IF 48T	MOVE	48 TO BB.	SVD04150
IF 49T	MOVE	49 TO BB.	SVD04160
IF 50T	MOVE	50 TO BB.	SVD04170
IF 51T	MOVE	51 TO BB.	SVD04180
IF 52T	MOVE	52 TO BB.	SVD04190
IF 53T	MOVE	53 TO BB.	SVD04200
IF 54T	MOVE	54 TO BB.	SVD04210
IF 55T	MOVE	55 TO AA.	SVD04220
IF 56T	MOVE	56 TO AA.	SVD04230
IF 57T	MOVE	57 TO BB.	SVD04240
IF 58T	MOVE	58 TO BB.	SVD04250
IF 59T	MOVE	59 TO BB.	SVD04260
IF 60T	MOVE	60 TO BB.	SVD04270
IF 61T	MOVE	61 TO BB.	SVD04280
IF 62T	MOVE	62 TO BB.	SVD04290
IF 63T	MOVE	63 TO BB.	SVD04300
IF 64T	MOVE	64 TO AA.	SVD04310
IF 65T	MOVE	65 TO BB.	SVD04320
IF 66T	MOVE	66 TO BB.	SVD04330
IF 67T	MOVE	67 TO AA.	SVD04340
IF 68T	MOVE	68 TO BB.	SVD04350
IF 69T	MOVE	69 TO BB.	SVD04360
IF 70T	MOVE	70 TO BB.	SVD04370
IF 71T	MOVE	71 TO BB.	SVD04380
IF 72T	MOVE	72 TO AA.	SVD04390
IF 73T	MOVE	73 TO BB.	SVD04400

IF 74T	MOVE 74 TO BB.	SVD04410
IF 75T	MOVE 75 TO BB.	SVD04420
IF 76T	MOVE 76 TO BB.	SVD04430
IF 77T	MOVE 77 TO BB.	SVD04440
IF 78T	MOVE 78 TO BB.	SVD04450
IF 79T	MOVE 79 TO BB.	SVD04460
IF 80T	MOVE 80 TO AA.	SVD04470
IF 81T	MOVE 81 TO BB.	SVD04480
IF 82T	MOVE 82 TO AA.	SVD04490
IF 83T	MOVE 83 TO BB.	SVD04500
IF 84T	MOVE 84 TO BB.	SVD04510
IF 85T	MOVE 85 TO AA.	SVD04520
IF 86T	MOVE 86 TO BB.	SVD04530
IF 87T	MOVE 87 TO BB.	SVD04540
IF 88T	MOVE 88 TO BB.	SVD04550
IF 89T	MOVE 89 TO BB.	SVD04560
IF 90T	MOVE 90 TO BB.	SVD04570
IF 91T	MOVE 91 TO BB.	SVD04580
IF 92T	MOVE 92 TO BB.	SVD04590
IF 93T	MOVE 93 TO AA.	SVD04600
IF 94T	MOVE 94 TO AA.	SVD04610
IF 95T	MOVE 95 TO AA.	SVD04620
IF 96T	MOVE 96 TO BB.	SVD04630
IF 97T	MOVE 97 TO BB.	SVD04640
IF 98T	MOVE 98 TO BB.	SVD04650
IF 99T	MOVE 99 TO BB.	SVD04660
IF 100T	MOVE 100 TO AA.	SVD04670
IF 101T	MOVE 101 TO BB.	SVD04680
IF 102T	MOVE 102 TO BB.	SVD04690
IF 103T	MOVE 103 TO BB.	SVD04700
IF 104T	MOVE 104 TO BB.	SVD04710
IF 105T	MOVE 105 TO AA.	SVD04720
IF 106T	MOVE 106 TO BB.	SVD04730
IF 107T	MOVE 107 TO AA.	SVD04740
IF 108T	MOVE 108 TO BB.	SVD04750
IF 109T	MOVE 109 TO BB.	SVD04760
IF 110T	MOVE 110 TO BB.	SVD04770
IF 111T	MOVE 111 TO BB.	SVD04780
IF 112T	MOVE 112 TO BB.	SVD04790
IF 113T	MOVE 113 TO BB.	SVD04800
IF 114T	MOVE 114 TO BB.	SVD04810
IF 115T	MOVE 115 TO BB.	SVD04820
IF 116T	MOVE 116 TO BB.	SVD04830
IF 117T	MOVE 117 TO AA.	SVD04840
IF 118T	MOVE 118 TO BB.	SVD04850
IF 119T	MOVE 119 TO BB.	SVD04860
IF 120T	MOVE 120 TO BB.	SVD04870
IF 121T	MOVE 121 TO BB.	SVD04880
IF 122T	MOVE 122 TO BB.	SVD04890
IF 123T	MOVE 123 TO BB.	SVD04900
IF 124T	MOVE 124 TO AA.	SVD04910
IF 125T	MOVE 125 TO BB.	SVD04920
IF 126T	MOVE 126 TO BB.	SVD04930
IF 127T	MOVE 127 TO BB.	SVD04940
IF 128T	MOVE 128 TO BB.	SVD04950



IF 129T	MOVE 129 TO BB.	SVD04960
IF 130T	MOVE 130 TO BB.	SVD04970
IF 131T	MOVE 131 TO AA.	SVD04980
IF 132T	MOVE 132 TO BB.	SVD04990
IF 133T	MOVE 133 TO BB.	SVD05000
IF 134T	MOVE 134 TO AA.	SVD05010
IF 135T	MOVE 135 TO BB.	SVD05020
IF 136T	MOVE 136 TO BB.	SVD05030
IF 137T	MOVE 137 TO AA.	SVD05040
IF 138T	MOVE 138 TO BB.	SVD05050
IF 139T	MOVE 139 TO AA.	SVD05060
IF 140T	MOVE 140 TO BB.	SVD05070
IF 141T	MOVE 141 TO AA.	SVD05080
IF 142T	MOVE 142 TO BB.	SVD05090
IF 143T	MOVE 143 TO BB.	SVD05100
IF 144T	MOVE 144 TO BB.	SVD05110
IF 145T	MOVE 145 TO AA.	SVD05120
IF 146T	MOVE 146 TO BB.	SVD05130
IF 147T	MOVE 147 TO BB.	SVD05140
IF 148T	MOVE 148 TO BB.	SVD05150
IF 149T	MOVE 149 TO BB.	SVD05160
IF 150T	MOVE 150 TO BB.	SVD05170
IF 151T	MOVE 151 TO BB.	SVD05180
IF 152T	MOVE 152 TO AA.	SVD05190
IF 153T	MOVE 153 TO BB.	SVD05200
IF 154T	MOVE 154 TO BB.	SVD05210
IF 155T	MOVE 155 TO BB.	SVD05220
IF 156T	MOVE 156 TO BB.	SVD05230
IF 157T	MOVE 157 TO BB.	SVD05240
IF 158T	MOVE 158 TO BB.	SVD05250
IF 159T	MOVE 159 TO AA.	SVD05260
IF 160T	MOVE 160 TO BB.	SVD05270
IF 161T	MOVE 161 TO BB.	SVD05280
IF 162T	MOVE 162 TO BB.	SVD05290
IF 163T	MOVE 163 TO BB.	SVD05300
IF 164T	MOVE 164 TO BB.	SVD05310
IF 165T	MOVE 165 TO BB.	SVD05320
IF 166T	MOVE 166 TO AA.	SVD05330
IF 167T	MOVE 167 TO BB.	SVD05340
IF 168T	MOVE 168 TO BB.	SVD05350
IF 169T	MOVE 169 TO BB.	SVD05360
IF 170T	MOVE 170 TO BB.	SVD05370
IF 171T	MOVE 171 TO BB.	SVD05380
IF 172T	MOVE 172 TO BB.	SVD05390
IF 173T	MOVE 173 TO BB.	SVD05400
IF 174T	MOVE 174 TO BB.	SVD05410
IF 175T	MOVE 175 TO AA.	SVD05420
IF 176T	MOVE 176 TO BB.	SVD05430
IF 177T	MOVE 177 TO BB.	SVD05440
IF 178T	MOVE 178 TO AA.	SVD05450
IF 179T	MOVE 179 TO BB.	SVD05460
IF 180T	MOVE 180 TO BB.	SVD05470
IF 181T	MOVE 181 TO BB.	SVD05480
IF 182T	MOVE 182 TO BB.	SVD05490
IF 183T	MOVE 183 TO BB.	SVD05500

IF 184T	MOVE 184 TO BB.	SVD05510
IF 185T	MOVE 185 TO BB.	SVD05520
IF 186T	MOVE 186 TO BB.	SVD05530
IF 187T	MOVE 187 TO BB.	SVD05540
IF 188T	MOVE 188 TO AA.	SVD05550
IF 189T	MOVE 189 TO BB.	SVD05560
IF 190T	MOVE 190 TO BB.	SVD05570
IF 191T	MOVE 191 TO BB.	SVD05580
IF 192T	MOVE 192 TO BB.	SVD05590
IF 193T	MOVE 193 TO AA.	SVD05600
IF 194T	MOVE 194 TO BB.	SVD05610
IF 195T	MOVE 195 TO BB.	SVD05620
IF 196T	MOVE 196 TO BB.	SVD05630
IF 197T	MOVE 197 TO BB.	SVD05640
IF 198T	MOVE 198 TO AA.	SVD05650
IF 199T	MOVE 199 TO BB.	SVD05660
IF 200T	MOVE 200 TO BB.	SVD05670
IF 201T	MOVE 201 TO BB.	SVD05680
IF 202T	MOVE 202 TO BB.	SVD05690
IF 203T	MOVE 203 TO BB.	SVD05700
IF 204T	MOVE 204 TO BB.	SVD05710
IF 205T	MOVE 205 TO BB.	SVD05720
IF 206T	MOVE 206 TO AA.	SVD05730
IF 207T	MOVE 207 TO BB.	SVD05740
IF 208T	MOVE 208 TO BB.	SVD05750
IF 209T	MOVE 209 TO BB.	SVD05760
IF 210T	MOVE 210 TO AA.	SVD05770
IF 211T	MOVE 211 TO BB.	SVD05780
IF 212T	MOVE 212 TO BB.	SVD05790
IF 213T	MOVE 213 TO BB.	SVD05800
IF 214T	MOVE 214 TO BB.	SVD05810
IF 215T	MOVE 215 TO BB.	SVD05820
IF 216T	MOVE 216 TO AA.	SVD05830
IF 217T	MOVE 217 TO AA.	SVD05840
IF 218T	MOVE 218 TO AA.	SVD05850
IF 219T	MOVE 219 TO BB.	SVD05860
IF 220T	MOVE 220 TO AA.	SVD05870
IF 221T	MOVE 221 TO BB.	SVD05880
IF 222T	MOVE 222 TO BB.	SVD05890
IF 223T	MOVE 223 TO BB.	SVD05900
IF 224T	MOVE 224 TO BB.	SVD05910
IF 225T	MOVE 225 TO BB.	SVD05920
IF 226T	MOVE 226 TO BB.	SVD05930
IF 227T	MOVE 227 TO BB.	SVD05940
IF 228T	MOVE 228 TO BB.	SVD05950
IF 229T	MOVE 229 TO AA.	SVD05960
IF 230T	MOVE 230 TO BB.	SVD05970
IF 231T	MOVE 231 TO BB.	SVD05980
IF 232T	MOVE 232 TO BB.	SVD05990
IF 233T	MOVE 233 TO BB.	SVD06000
IF 234T	MOVE 234 TO BB.	SVD06010
IF 235T	MOVE 235 TO BB.	SVD06020
IF 236T	MOVE 236 TO BB.	SVD06030
IF 237T	MOVE 237 TO BB.	SVD06040
IF 238T	MOVE 238 TO AA.	SVD06050

IF 239T	MOVE 239 TO BB.	SVD06060
IF 240T	MOVE 240 TO BB.	SVD06070
IF 241T	MOVE 241 TO BB.	SVD06080
IF 242T	MOVE 242 TO BB.	SVD06090
IF 243T	MOVE 243 TO BB.	SVD06100
IF 244T	MOVE 244 TO BB.	SVD06110
IF 245T	MOVE 245 TO AA.	SVD06120
IF 246T	MOVE 246 TO BB.	SVD06130
IF 247T	MOVE 247 TO BB.	SVD06140
IF 248T	MOVE 248 TO BB.	SVD06150
IF 249T	MOVE 249 TO BB.	SVD06160
IF 250T	MOVE 250 TO BB.	SVD06170
IF 251T	MOVE 251 TO BB.	SVD06180
IF 252T	MOVE 252 TO BB.	SVD06190
IF 253T	MOVE 253 TO BB.	SVD06200
IF 254T	MOVE 254 TO AA.	SVD06210
IF 255T	MOVE 255 TO BB.	SVD06220
IF 256T	MOVE 256 TO BB.	SVD06230
IF 257T	MOVE 257 TO BB.	SVD06240
IF 258T	MOVE 258 TO BB.	SVD06250
IF 259T	MOVE 259 TO BB.	SVD06260
IF 260T	MOVE 260 TO BB.	SVD06270
IF 261T	MOVE 261 TO BB.	SVD06280
IF 262T	MOVE 262 TO AA.	SVD06290
IF 263T	MOVE 263 TO AA.	SVD06300
IF 264T	MOVE 264 TO BB.	SVD06310
IF 265T	MOVE 265 TO BB.	SVD06320
IF 266T	MOVE 266 TO AA.	SVD06330
IF 267T	MOVE 267 TO BB.	SVD06340
IF 268T	MOVE 268 TO BB.	SVD06350
IF 269T	MOVE 269 TO BB.	SVD06360
IF 270T	MOVE 270 TO AA.	SVD06370
IF 271T	MOVE 271 TO AA.	SVD06380
IF 272T	MOVE 272 TO BB.	SVD06390
IF 273T	MOVE 273 TO BB.	SVD06400
IF 274T	MOVE 274 TO AA.	SVD06410
IF 275T	MOVE 275 TO BB.	SVD06420
IF 276T	MOVE 276 TO AA.	SVD06430
IF 277T	MOVE 277 TO AA.	SVD06440
IF 278T	MOVE 278 TO AA.	SVD06450
IF 279T	MOVE 279 TO EAA.	SVD06460
IF 280T	MOVE 280 TO EBB.	SVD06470
IF 281T	MOVE 281 TO EBB.	SVD06480
IF 282T	MOVE 282 TO EBB.	SVD06490
IF 283T	MOVE 283 TO EBB.	SVD06500
IF 284T	MOVE 284 TO EBB.	SVD06510
IF 285T	MOVE 285 TO EAA.	SVD06520
IF 286T	MOVE 286 TO EBB.	SVD06530
IF 287T	MOVE 287 TO EBB.	SVD06540
IF 288T	MOVE 288 TO EBB.	SVD06550
IF 289T	MOVE 289 TO EAA.	SVD06560
IF 290T	MOVE 290 TO EAA.	SVD06570
IF 291T	MOVE 291 TO EAA.	SVD06580
IF 292T	MOVE 292 TO EAA.	SVD06590
IF 293T	MOVE 293 TO EBB.	SVD06600

```

        IF 294T      MOVE 294 TO EBB.          SVD06610
        IF 295T      MOVE 295 TO EBB.          SVD06620
        IF 296T      MOVE 296 TO EBB.          SVD06630
        IF 297T      MOVE 297 TO EBB.          SVD06640
        IF 298T      MOVE 298 TO EAA.          SVD06650
        IF 299T      MOVE 299 TO EAA.          SVD06660
        IF 300T      MOVE 300 TO EAA.          SVD06670
        IF 301T      MOVE 301 TO EAA.          SVD06680
        IF 302T      MOVE 302 TO EBB.          SVD06690
        IF 303T      MOVE 303 TO EBB.          SVD06700
        MOVE DBICYO TO TTT.                    SVD06710
        WRITE TA-R.                             SVD06720
        MOVE SPACE TO TA-R.                     SVD06730
        ADD 1 TO OCNT.                          SVD06740
        GO TO READ-RTN.                         SVD06750
    END-RTN.                                    SVD06760
        DISPLAY "   ECNT = " ECNT.             SVD06770
        DISPLAY "   OCNT = " OCNT.             SVD06780
        CLOSE T-F TA-F.                        SVD06790
        OPEN INPUT TA-F.                       SVD06800
    RD-RTN.                                     SVD06810
        READ TA-F AT END CLOSE TA-F STOP RUN.  SVD06820
        DISPLAY " " TA-R.                      SVD06830
    PER-RTN.                                    SVD06840
        PERFORM RD-RTN 700 TIMES.              SVD06850
        CLOSE TA-F.                            SVD06860
        STOP RUN.                              SVD06870
/*                                             SVD06880
//GO.TAPEIN DD DSN=SVBWED,UNIT=TAPE,DISP=OLD,VOL=SER=SNOW14, SVD06890
//          DCB=EROPT=SKP,LABEL=(2,BLF)       SVD06900
//GO.TAPEOUT DD DSN=SNOW11,UNIT=TAPE,DISP=(NEW,KEEP), SVD06910
//          VOL=(, ,99,SER=SCRATCH)           SVD06920
//SYSOUT DD SYSOUT=A                          SVD06930
//                                             SVD06940

```



' 020	,	SVD00560
' 023	,	SVD00570
' 030	,	SVD00580
' 032	,	SVD00590
' 033	,	SVD00600
' 034, 035	,	SVD00610
' 036	,	SVD00620
' 037	,	SVD00630
' 038	,	SVD00640
' *045 - 079	,	SVD00650
' 045	,	SVD00660
' 050	,	SVD00670
' 055	,	SVD00680
' 056	,	SVD00690
' 060	,	SVD00700
' 062 - 064	,	SVD00710
' 070	,	SVD00720
' 071	,	SVD00730
' 076	,	SVD00740
' *080 - 088	,	SVD00750
' 080	,	SVD00760
' 081 - 083	,	SVD00770
' 084	,	SVD00780
' 085	,	SVD00790
' 086	,	SVD00800
' *090 - 099	,	SVD00810
' 090 - 097	,	SVD00820
' 098	,	SVD00830
' *100 - 139	,	SVD00840
' 100 - 104	,	SVD00850
' 110 - 118	,	SVD00860
' 120	,	SVD00870
' 122	,	SVD00880
' 125	,	SVD00890
' 126	,	SVD00900
' 121, 123, 124, 127-129	,	SVD00910
' 137	,	SVD00920
' 138	,	SVD00930
' *140 - 149	,	SVD00940
' *150 - 159	,	SVD00950
' 150	,	SVD00960
' 151	,	SVD00970
' 152	,	SVD00980
' 153	,	SVD00990
' 154	,	SVD01000
' 155.0	,	SVD01010
' 157	,	SVD01020
' *160 - 165	,	SVD01030
' 161	,	SVD01040
' 162	,	SVD01050
' *170 - 175	,	SVD01060
' 170	,	SVD01070
' 172	,	SVD01080
' 173	,	SVD01090
' 174	,	SVD01100

'*179 - 189	'	SVD01110
' 180	'	SVD01120
' 181	'	SVD01130
' 179,182	'	SVD01140
' 183	'	SVD01150
' 185	'	SVD01160
' 184	'	SVD01170
' 188	'	SVD01180
'*190 - 199	'	SVD01190
' 191	'	SVD01200
'*200 - 208	'	SVD01210
' 201	'	SVD01220
' 204 - 208	'	SVD01230
'*210 - 229	'	SVD01240
' 214	'	SVD01250
' 217	'	SVD01260
' 218,219	'	SVD01270
' 220	'	SVD01280
' 223	'	SVD01290
' 225	'	SVD01300
' 226	'	SVD01310
'*230 - 234	'	SVD01320
'*235 - 239	'	SVD01330
'*240-259,270-279	'	SVD01340
' 240 - 244	'	SVD01350
' 250	'	SVD01360
' 272.0-272.4	'	SVD01370
' 278.0	'	SVD01380
'*260-269	'	SVD01390
' 260	'	SVD01400
' 261	'	SVD01410
' 262,263	'	SVD01420
' 264-269	'	SVD01430
'*280-289	'	SVD01440
' 280-285	'	SVD01450
'*290-319	'	SVD01460
' 290	'	SVD01470
' 295	'	SVD01480
' 296	'	SVD01490
' 291-294,297-299	'	SVD01500
' 300,301	'	SVD01510
' 303	'	SVD01520
' 304	'	SVD01530
' 306	'	SVD01540
' 317 - 319	'	SVD01550
'*320 - 359	'	SVD01560
' 320 - 322	'	SVD01570
' 332	'	SVD01580
' 330,331,333-336	'	SVD01590
' 340	'	SVD01600
' 343,344	'	SVD01610
' 345	'	SVD01620
'*360 - 379	'	SVD01630
' 365	'	SVD01640
' 366	'	SVD01650

' 369	' ,	SVD01660
' 372.0-372.3	' ,	SVD01670
' 375	' ,	SVD01680
' 378	' ,	SVD01690
' *380 - 389	' ,	SVD01700
' 381 - 383	' ,	SVD01710
' 389	' ,	SVD01720
' *390 - 398	' ,	SVD01730
' 390 - 392	' ,	SVD01740
' 393 - 398	' ,	SVD01750
' *401 - 405	' ,	SVD01760
' 402,404	' ,	SVD01770
' *410 - 414	' ,	SVD01780
' 410	' ,	SVD01790
' *415 - 429	' ,	SVD01800
' 415.1	' ,	SVD01810
' 427	' ,	SVD01820
' 427.5	' ,	SVD01830
' *430 - 438	' ,	SVD01840
' 430	' ,	SVD01850
' 431,432	' ,	SVD01860
' 433,434	' ,	SVD01870
' 436	' ,	SVD01880
' 437.0	' ,	SVD01890
' 437.8	' ,	SVD01900
' *440 - 459	' ,	SVD01910
' 440	' ,	SVD01920
' 444	' ,	SVD01930
' 441-443,446-448	' ,	SVD01940
' 451 - 453	' ,	SVD01950
' 454	' ,	SVD01960
' 455	' ,	SVD01970
' *460-465,470-478	' ,	SVD01980
' 463	' ,	SVD01990
' 464	' ,	SVD02000
' 460-462,465	' ,	SVD02010
' 470,471	' ,	SVD02020
' 472,473	' ,	SVD02030
' 474	' ,	SVD02040
' *466,480-519	' ,	SVD02050
' 466	' ,	SVD02060
' 480 - 486	' ,	SVD02070
' 487	' ,	SVD02080
' 490 - 493	' ,	SVD02090
' 494	' ,	SVD02100
' 495,496	' ,	SVD02110
' 500 - 508	' ,	SVD02120
' 511	' ,	SVD02130
' *320 - 529	' ,	SVD02140
' 520 - 525	' ,	SVD02150
' 526	' ,	SVD02160
' *530 - 579	' ,	SVD02170
' 530	' ,	SVD02180
' 531 - 533	' ,	SVD02190
' 540 - 543	' ,	SVD02200



' 550 - 553	' ,	SVD02210
' 560	' ,	SVD02220
' 562	' ,	SVD02230
' 564	' ,	SVD02240
' 571	' ,	SVD02250
' 574-575.1	' ,	SVD02260
' *580 - 599	' ,	SVD02270
' 580 - 589	' ,	SVD02280
' 590	' ,	SVD02290
' 592,594	' ,	SVD02300
' 595	' ,	SVD02310
' *600 - 608	' ,	SVD02320
' 600	' ,	SVD02330
' 603	' ,	SVD02340
' 605	' ,	SVD02350
' 606	' ,	SVD02360
' *610 - 629	' ,	SVD02370
' 610,611	' ,	SVD02380
' 614.0-614.2	' ,	SVD02390
' 614.3-614.9	' ,	SVD02400
' 615,616	' ,	SVD02410
' 618	' ,	SVD02420
' 626.0-626.5	' ,	SVD02430
' 628	' ,	SVD02440
' *630 - 639	' ,	SVD02450
' 634	' ,	SVD02460
' 635	' ,	SVD02470
' 636	' ,	SVD02480
' *640-646,651-676	' ,	SVD02490
' 640,641,666	' ,	SVD02500
' 642.4-642.9,643	' ,	SVD02510
' 644.6	' ,	SVD02520
' 660	' ,	SVD02530
' 670 - 676	' ,	SVD02540
' *647,648	' ,	SVD02550
' *650	' ,	SVD02560
' *680 - 709	' ,	SVD02570
' 680 - 686	' ,	SVD02580
' *710 - 739	' ,	SVD02590
' 714	' ,	SVD02600
' 710-713,715,716	' ,	SVD02610
' 717 - 719	' ,	SVD02620
' 720.0	' ,	SVD02630
' 720.1-724	' ,	SVD02640
' 725 - 729	' ,	SVD02650
' 730	' ,	SVD02660
' 734 - 736	' ,	SVD02670
' *740 - 759	' ,	SVD02680
' 741,742.3	' ,	SVD02690
' 740,742.0-2,742.4-9	' ,	SVD02700
' 745 - 747	' ,	SVD02710
' 749	' ,	SVD02720
' 750,751	' ,	SVD02730
' 752.5	' ,	SVD02740
' 754.3	' ,	SVD02750

' 754.0-2,754.4-756	' ,	SVD02760
' *760 - 779	' ,	SVD02770
' 760	' ,	SVD02780
' 761 - 763	' ,	SVD02790
' 764,765	' ,	SVD02800
' 767	' ,	SVD02810
' 768 - 770	' ,	SVD02820
' 773	' ,	SVD02830
' *780 - 799	' ,	SVD02840
' 780.6	' ,	SVD02850
' 785.0-785.3	' ,	SVD02860
' 788.0	' ,	SVD02870
' 788.2	' ,	SVD02880
' 789.0	' ,	SVD02890
' 797	' ,	SVD02900
' 798.0	' ,	SVD02910
' 799.1	' ,	SVD02920
' *800 - 829	' ,	SVD02930
' 800 - 804	' ,	SVD02940
' 805 - 809	' ,	SVD02950
' 812,813	' ,	SVD02960
' 820	' ,	SVD02970
' 821	' ,	SVD02980
' 823,824	' ,	SVD02990
' 810-1,814-9,822,5,9	' ,	SVD03000
' *830 - 848	' ,	SVD03010
' *850-869,950-957	' ,	SVD03020
' 850	' ,	SVD03030
' 851-854,950-951	' ,	SVD03040
' *870 - 904	' ,	SVD03050
' 870 - 873	' ,	SVD03060
' 880 - 887	' ,	SVD03070
' 890 - 897	' ,	SVD03080
' *930 - 939	' ,	SVD03090
' *940 - 949	' ,	SVD03100
' 940	' ,	SVD03110
' 944	' ,	SVD03120
' *960 - 989	' ,	SVD03130
' 960 - 979	' ,	SVD03140
' *996 - 999	' ,	SVD03150
' *910-929,958-9,990-5	' ,	SVD03160
' *905 - 909	' ,	SVD03170
' *E800 - E848	' ,	SVD03180
' E800 - E807	' ,	SVD03190
' E810 - E819	' ,	SVD03200
' E826 - E829	' ,	SVD03210
' E830 - E838	' ,	SVD03220
' E840 - E845	' ,	SVD03230
' *E850 - E869	' ,	SVD03240
' E850 - E858	' ,	SVD03250
' E860 - E866	' ,	SVD03260
' E867 - E869	' ,	SVD03270
' *E870 - E879	' ,	SVD03280
' *E880 - E888	' ,	SVD03290
' *E890 - E899	' ,	SVD03300

```

          '*E900 - E929          ', SVD03310
          ' E900 - E909          ', SVD03320
          ' E910                  ', SVD03330
          ' E914,E915            ', SVD03340
          ' E919,E920            ', SVD03350
          ' E922                  ', SVD03360
          '*E930 - E949          ', SVD03370
          '*E950 - E959          ', SVD03380
          '*E960 - E969          ', SVD03390
          '*E970 - E979          ', SVD03400
          ' E980 - E989          ', SVD03410
          ' E990 - E999          ') ; SVD03420
DCL MF CHAR(2); SVD03430
DCL FSW PIC '9' INIT(0); SVD03440
DCL BUN CHAR(21); SVD03450
DCL HJ CHAR(6); SVD03460
DCL TCNT PIC '(6)9' INIT(0); SVD03470
DCL OCNT PIC '(6)9' INIT(0); SVD03480
DCL ICNT PIC '(6)9' INIT(0); SVD03490
DCL PCNT PIC '(6)9' INIT(0); SVD03500
OPEN FILE(TF),FILE(SYSPRINT) LINESIZE(132) PAGESIZE(120); SVD03510
ON ENDFILE(TF) GO TO GROSS; SVD03520
AR = 0; SVD03530
RD: READ FILE(TF) INTO(TR); SVD03540
    TCNT = TCNT + 1 ; SVD03550
    IF SUBSTR(DEACA,1,3) = '000' THEN GOTO RD; SVD03560
NN: DO N = 1 TO 5; SVD03570
    IF B(N) ^= 0 THEN SVD03580
        CALL DEATH; SVD03590
    END; SVD03600
    CALL TOTAL; SVD03610
    GOTO RD; SVD03620
DEATH: PROC; SVD03630
/* AMOUNT OF DEATH BY SEXES & AGE & NO OF I.C.D(307) 5S(4) */ SVD03640
    IF AGY < 1 THEN L = 2; SVD03650
    ELSE IF AGY < 5 THEN L = 3; SVD03660
    ELSE IF AGY < 10 THEN L = 4; SVD03670
    ELSE IF AGY < 15 THEN L = 5; SVD03680
    ELSE IF AGY < 20 THEN L = 6; SVD03690
    ELSE IF AGY < 25 THEN L = 7; SVD03700
    ELSE IF AGY < 30 THEN L = 8; SVD03710
    ELSE IF AGY < 35 THEN L = 9; SVD03720
    ELSE IF AGY < 40 THEN L = 10; SVD03730
    ELSE IF AGY < 45 THEN L = 11; SVD03740
    ELSE IF AGY < 50 THEN L = 12; SVD03750
    ELSE IF AGY < 55 THEN L = 13; SVD03760
    ELSE IF AGY < 60 THEN L = 14; SVD03770
    ELSE IF AGY < 65 THEN L = 15; SVD03780
    ELSE IF AGY < 70 THEN L = 16; SVD03790
    ELSE IF AGY < 75 THEN L = 17; SVD03800
    ELSE IF AGY < 80 THEN L = 18; SVD03810
    ELSE IF AGY < 85 THEN L = 19; SVD03820
    ELSE L = 20; SVD03830
    IF DJD = '1' THEN SVD03840
        AR(1,B(N)+1,DSEX+1,L)=AR(1,B(N)+1,DSEX+1,L)+1; SVD03850

```

```

                AR(2,B(N)+1,DSEX+1,L)=AR(2,B(N)+1,DSEX+1,L)+1; SVD03860
IF DJGUN < 31 THEN SVD03870
                AR(3,B(N)+1,DSEX+1,L)=AR(3,B(N)+1,DSEX+1,L)+1; SVD03880
IF DJDO = '11'| DJDO = '21'| DJDO= '22'| DJDO = '23' THEN SVD03890
                AR(4,B(N)+1,DSEX+1,L)=AR(4,B(N)+1,DSEX+1,L)+1; SVD03900
END DEATH; SVD03910
TOTAL : PROC; SVD03920
        IF AGY < 1 THEN L = 2; SVD03930
        ELSE IF AGY < 5 THEN L = 3; SVD03940
        ELSE IF AGY < 10 THEN L = 4; SVD03950
        ELSE IF AGY < 15 THEN L = 5; SVD03960
        ELSE IF AGY < 20 THEN L = 6; SVD03970
        ELSE IF AGY < 25 THEN L = 7; SVD03980
        ELSE IF AGY < 30 THEN L = 8; SVD03990
        ELSE IF AGY < 35 THEN L = 9; SVD04000
        ELSE IF AGY < 40 THEN L = 10; SVD04010
        ELSE IF AGY < 45 THEN L = 11; SVD04020
        ELSE IF AGY < 50 THEN L = 12; SVD04030
        ELSE IF AGY < 55 THEN L = 13; SVD04040
        ELSE IF AGY < 60 THEN L = 14; SVD04050
        ELSE IF AGY < 65 THEN L = 15; SVD04060
        ELSE IF AGY < 70 THEN L = 14; SVD04070
        ELSE IF AGY < 75 THEN L = 17; SVD04080
        ELSE IF AGY < 80 THEN L = 18; SVD04090
        ELSE IF AGY < 85 THEN L = 19; SVD04100
        ELSE L = 20; SVD04110
IF DJD = '1' THEN SVD04120
                AR(1,1,DSEX+1,L)=AR(1,1,DSEX+1,L)+1; SVD04130
                AR(2,1,DSEX+1,L)=AR(2,1,DSEX+1,L)+1; SVD04140
IF DJGUN < 31 THEN SVD04150
                AR(3,1,DSEX+1,L)=AR(3,1,DSEX+1,L)+1; SVD04160
IF DJDO = '11'| DJDO = '21'| DJDO= '22'| DJDO = '23' THEN SVD04170
                AR(4,1,DSEX+1,L)=AR(4,1,DSEX+1,L)+1; SVD04180
END TOTAL; SVD04190
GROSS: SVD04200
        DO K = 2 TO 21; SVD04210
                AR(*,*,*,1) = AR(*,*,*,1) + AR(*,*,*,K); SVD04220
        END; SVD04230
                AR(*,*,1,*) = AR(*,*,2,*) + AR(*,*,3,*); SVD04240
PRT: SVD04250
        DO I = 1 TO 4; SVD04260
                DO J = 1 TO 304; SVD04270
                        IF J = 145 | J = 152 THEN GOTO J11; SVD04280
                        IF J = 1|J = 13| J = 25| J = 37| J = 49| J = 61| J = 73| SVD04290
                                J = 85| J = 97| J =109| J =121| J =133| J =146| SVD04300
                                J =159| J =171| J =183| J =195| J =207| J =219| SVD04310
                                J =231| J =243| J =255| J =267| J =279| J =292 SVD04320
                        THEN DO; PUT PAGE; SVD04330
                                CALL AG; SVD04340
                        END; SVD04350
                        BUN= MOT(J); SVD04360
                        DO K = 1 TO 3; SVD04370
                                CALL TSEX; SVD04380
                                IF K = 2 | K = 3 THEN BUN = (21) ' '; SVD04390
                                PUT SKIP(2) EDIT(BUN,MF,(AR(I,J,K,L) DO L = 1 TO 10)) SVD04400

```

```

                                (A,X(20),A,10 F(8));
                                SVD04410
                                END;
                                SVD04420
J11: END;
                                SVD04430
                                END;
                                SVD04440
                                PUT PAGE;
                                SVD04450
                                DO I = 1 TO 4;
                                SVD04460
                                    FSW= 1;
                                    SVD04470
                                DO J = 1 TO 304;
                                SVD04480
                                    IF J = 145 | J = 152 THEN GOTO J22;
                                    SVD04490
                                    IF J = 1 | J = 13 | J = 25 | J = 37 | J = 49 | J = 61 | J = 73 |
                                SVD04500
                                        J = 85 | J = 97 | J = 109 | J = 121 | J = 133 | J = 146 |
                                SVD04510
                                        J = 159 | J = 171 | J = 183 | J = 195 | J = 207 | J = 219 |
                                SVD04520
                                        J = 231 | J = 243 | J = 255 | J = 267 | J = 279 | J = 292
                                SVD04530
                                    THEN DO; PUT PAGE;
                                SVD04540
                                        CALL AG;
                                SVD04550
                                    END;
                                SVD04560
                                BUN= NOT(J);
                                SVD04570
                                DO K = 1 TO 3;
                                SVD04580
                                    CALL TSEX;
                                SVD04590
                                    IF K = 2 | K = 3 THEN BUN = (2) ' ' ;
                                SVD04600
                                    PUT SKIP(2) EDIT((AR(I,J,K,L) DO L = 11 TO 20),MF,BUN)
                                SVD04610
                                        (10 F(8),X(3),A,X(25),A);
                                SVD04620
                                END;
                                SVD04630
                                J22: END;
                                SVD04640
                                END;
                                SVD04650
                                TSEX: PROC;
                                SVD04660
                                    IF K = 1 THEN MF = ' T ' ;
                                SVD04670
                                    ELSE IF K = 2 THEN MF = ' M ' ;
                                SVD04680
                                    ELSE IF K = 3 THEN MF = ' F ' ;
                                SVD04690
                                END TSEX;
                                SVD04700
                                AG: PROC;
                                SVD04710
                                    IF FSW = 0 THEN
                                SVD04720
                                        DO; PUT SKIP EDIT(' THE CAUSE OF DEATH ' , ' S ' ,
                                SVD04730
                                            ' TOT ' , ' -1 ' ,
                                SVD04740
                                            ' 1-4 ' , ' 5-9 ' , ' 10-14 ' , ' 15-19 ' , ' 20-24 ' ,
                                SVD04750
                                            ' 25-29 ' , ' 30-34 ' , ' 35-39 ' )
                                SVD04760
                                            (A,X(20),A,10 A);
                                SVD04770
                                            PUT SKIP EDIT((132)'=' )(A);
                                SVD04780
                                        END;
                                SVD04790
                                    ELSE DO;
                                SVD04800
                                        PUT SKIP(4) EDIT(' 40-44 ' , ' 45-49 ' , ' 50-54 ' ,
                                SVD04810
                                            ' 55-59 ' , ' 60-64 ' , ' 65-69 ' , ' 70-74 ' ,
                                SVD04820
                                            ' 75-79 ' , ' 80-84 ' , ' 85- ' ,
                                SVD04830
                                            ' S ' , ' THE CAUSE OF DEATH ' )
                                SVD04840
                                            (10 A,X(3),A,X(25),A);
                                SVD04850
                                        PUT SKIP EDIT((132)'=' )(A);
                                SVD04860
                                    END;
                                SVD04870
                                END AG;
                                SVD04880
                                LAST:
                                SVD04890
                                    PUT PAGE;
                                SVD04900
                                    PUT SKIP EDIT('INPUT COUNT =',TCNT)(A,A);
                                SVD04910
                                    PUT SKIP EDIT('PYOBON COUNT =',OCNT)(A,A);
                                SVD04920
                                    PUT SKIP EDIT('MYUNGBUCOUNT =',ICNT)(A,A);
                                SVD04930
                                    CLOSE FILE(TF);
                                SVD04940
                                END GOLD;
                                SVD04950

```

FILE: SVDWTZ14 TAB307 A1 VM/SP RELEASE 2.1 EXPRESS PUT8302+ SLU203

/K		SVD04960
//GO.TF	DD UNIT=TAPE,DSN=HH,VOL=SER=123321, LABEL=(2,BLP),	SVD04970
//	DCB=(RECFM=FB,BLKSIZE=3200,LRECL=80),DISP=(OLD,KEEP)	SVD04980
//GO.SYSOUT	DD SYSOUT=A	SVD04990
//GO.SYSPRINT	DD SYSOUT=A	SVD05000
//		SVD05010

```

//SVDWS914 JOB CLASS=A, TYPRUN=HOLD
//STEP1 EXEC PLIFCLG
//PLI.SYSIN DD *
* PROCESS GS, NEST, OPT(2) ;
  GOLD: PROC OPTIONS(MAIN) ;
    DCL TF1 FILE RECORD INPUT;
    DCL 1 TR1,
        2 DKEY CHAR(16),
        2 DBDO CHAR(2),
        2 DBGUN CHAR(2),
        2 DJDO CHAR(2),
        2 DJGUN CHAR(2),
        2 DSEX PIC '(1)9',
        2 DBYR PIC '(3)9',
        2 DBMN PIC '(2)9',
        2 DBDYY PIC '(2)9',
        2 DDYR PIC '(3)9',
        2 DDMN PIC '(2)9',
        2 DDDYY PIC '(2)9',
        2 DPLS PIC '(1)9',
        2 DOPT PIC '(2)9',
        2 DJD PIC '(1)9',
        2 DMST PIC '(1)9',
        2 DEDT PIC '(1)9',
        2 DEACA CHAR(9),
        2 AGY PIC '(3)9',
        2 AGW PIC '(2)9',
        2 AGD PIC '(2)9',
        2 B(5) PIC '(3)9',
        2 F1 CHAR(4);
    DCL TF2 FILE RECORD INPUT;
    DCL RF FILE RECORD OUTPUT;
    DCL 1 RR,
        2 PYO CHAR(1) INIT(' '),
        2 ECD CHAR(1) INIT(' '),
        2 ICD CHAR(3) INIT(' '),
        2 SEX CHAR(1) INIT(' '),
        2 AGE(21) PIC '(6)9' INIT((21)0);
    DCL AR0(2,3,21) PIC '(6)9';
    DCL AR1(2,3,21) PIC '(6)9';
    DCL I PIC '9' INIT(0);
    DCL J PIC '9' INIT(0);
    DCL K PIC '(3)9' INIT(0);
    DCL L PIC '(3)9' INIT(0);
    DCL TCNT PIC '(6)9' INIT(0);
    DCL OCNT PIC '(6)9' INIT(0);
    DCL AA CHAR(3) INIT(' ');
    DCL EEE PIC '9' INIT(0);
    OPEN FILE(TF1), FILE(RF);
    ON ENDFILE(TF1) GO TO GROSS;
    ON ENDFILE(TF2) GO TO GROSS;
    AR0 = 0;
    AR1 = 0;
    RD: IF EEE = 0 THEN READ FILE(TF1) INTO(TR1);
        ELSE READ FILE(TF2) INTO(TR1);

```

SVD00010  
SVD00020  
SVD00030  
SVD00040  
SVD00050  
SVD00060  
SVD00070  
SVD00080  
SVD00090  
SVD00100  
SVD00110  
SVD00120  
SVD00130  
SVD00140  
SVD00150  
SVD00160  
SVD00170  
SVD00180  
SVD00190  
SVD00200  
SVD00210  
SVD00220  
SVD00230  
SVD00240  
SVD00250  
SVD00260  
SVD00270  
SVD00280  
SVD00290  
SVD00300  
SVD00310  
SVD00320  
SVD00330  
SVD00340  
SVD00350  
SVD00360  
SVD00370  
SVD00380  
SVD00390  
SVD00400  
SVD00410  
SVD00420  
SVD00430  
SVD00440  
SVD00450  
SVD00460  
SVD00470  
SVD00480  
SVD00490  
SVD00500  
SVD00510  
SVD00520  
SVD00530  
SVD00540  
SVD00550

```

IF EEE = 0 & SUBSTR(DEACA,1,3) = '000' THEN GOTO RD;
IF EEE = 1 & SUBSTR(DEACA,6,3) = '000' THEN GOTO RD;
TCNT = TCNT + 1;
IF TCNT = 1 THEN DO;
    IF EEE = 0
        THEN AA = SUBSTR(DEACA,1,3);
    ELSE AA = SUBSTR(DEACA,6,3); END;
IF EEE = 0 THEN DO;
    IF AA ^= SUBSTR(DEACA,1,3) THEN
        DO; CALL CRT;
        AA = SUBSTR(DEACA,1,3);
        END;
    END;
ELSE DO;
    IF AA ^= SUBSTR(DEACA,6,3) THEN
        DO; CALL CRT;
        AA = SUBSTR(DEACA,6,3);
        END;
    END;
END;
ACCUM: IF AGY < 1 THEN K = 2;
ELSE IF AGY < 5 THEN K = 3;
ELSE IF AGY < 10 THEN K = 4;
ELSE IF AGY < 15 THEN K = 5;
ELSE IF AGY < 20 THEN K = 6;
ELSE IF AGY < 25 THEN K = 7;
ELSE IF AGY < 30 THEN K = 8;
ELSE IF AGY < 35 THEN K = 9;
ELSE IF AGY < 40 THEN K = 10;
ELSE IF AGY < 45 THEN K = 11;
ELSE IF AGY < 50 THEN K = 12;
ELSE IF AGY < 55 THEN K = 13;
ELSE IF AGY < 60 THEN K = 14;
ELSE IF AGY < 65 THEN K = 15;
ELSE IF AGY < 70 THEN K = 16;
ELSE IF AGY < 75 THEN K = 17;
ELSE IF AGY < 80 THEN K = 18;
ELSE IF AGY < 85 THEN K = 19;
ELSE K = 20;
AR1(1,DSEX+1,K) = AR1(1,DSEX+1,K) + 1;
IF DJD = '1' THEN
    AR1(2,DSEX+1,K) = AR1(2,DSEX+1,K) + 1;
GOTO RD;
CRT: PROC;
ARO = ARO + AR1;
AR1(*,1,*) = AR1(*,2,*) + AR1(*,3,*) ;
DO K = 2 TO 21;
    AR1(*,*,1) = AR1(*,*,1) + AR1(*,*,K);
END;
DO I = 1 TO 2;
    DO J = 1 TO 3;
        DO K = 1 TO 21;
            AGE(K) = AR1(I,J,K);
        END;
    PYO = I;

```



```

SEX = J;
IF EEE = 0 THEN ECD = '1';
ELSE ECD = '2';
ICD = AA;
WRITE FILE(RF) FROM(RR);
OCNT = OCNT + 1;
END;
END;
ARI = 0;
END CRT;
GROSS:
CALL CRT;
ARI = AR0; AA = ' '; CALL CRT;
LAST1:
IF EEE = 0 THEN DO;
CLOSE FILE(TF1);
OPEN FILE(TF2);
TCNT = 0; AR0 = 0; ARI = 0;
EEE = 1;
GOTO RD;
END;
LAST2:
PUT SKIP EDIT('INPUT COUNT =',TCNT)(A,A);
PUT SKIP EDIT('SUMMR COUNT =',OCNT)(A,A);
PUT SKIP EDIT('MYUNGBUCOUNT =',ICNT)(A,A);
CLOSE FILE(TF2),FILE(RF);
END GOLD;
/*
//GO.TF1 DD UNIT=TAPE,DSN=HH,VOL=SER=NN3977,LABEL=(2,BLP),
// DCB=(RECFM=FB,BLKSIZE=3200,LRECL=80),DISP=(OLD,KEEP)
//GO.TF2 DD UNIT=TAPE,DSN=HB,VOL=SER=NN3473,LABEL=(2,BLP),
// DCB=(RECFM=FB,BLKSIZE=3200,LRECL=80),DISP=(OLD,KEEP)
//GO.RF DD UNIT=TAPE,DSN=HO,DISP=(NEW,KEEP),
// DCB=(RECFM=FB,BLKSIZE=3960,LRECL=132)
//GO.SYSOUT DD SYSOUT=A
//GO.SYSPRINT DD SYSOUT=A
//

```

SVD01110  
SVD01120  
SVD01130  
SVD01140  
SVD01150  
SVD01160  
SVD01170  
SVD01180  
SVD01190  
SVD01200  
SVD01210  
SVD01220  
SVD01230  
SVD01240  
SVD01250  
SVD01260  
SVD01270  
SVD01280  
SVD01290  
SVD01300  
SVD01310  
SVD01320  
SVD01330  
SVD01340  
SVD01350  
SVD01360  
SVD01370  
SVD01380  
SVD01390  
SVD01400  
SVD01410  
SVD01420  
SVD01430  
SVD01440  
SVD01450  
SVD01460  
SVD01470

```

//SVDWT914 JOB CLASS=A,TYPRUN=HOLD
//STEP2 EXEC PLIFCLG
//PLI.SYSIN DD *
* PROCESS GS,NEST,OPT(2) ;
  GOLD: PROC OPTIONS(MAIN) ;
    DCL RF FILE RECORD INPUT;
    DCL I RR,
        2 PYD CHAR(1) INIT(' '),
        2 ECD CHAR(1) INIT(' '),
        2 ICD CHAR(3) INIT(' '),
        2 SEX CHAR(1) INIT(' '),
        2 AGE(21) PIC '(6)9' INIT((21)0);
    DCL I PIC '9' INIT(0);
    DCL J PIC '9' INIT(0);
    DCL K PIC '(3)9' INIT(0);
    DCL L PIC '(3)9' INIT(0);
    DCL TCNT PIC '(6)9' INIT(0);
    DCL LCNT PIC '(6)9' INIT(0);
    DCL AA CHAR(3) INIT(' ');
    DCL EEE PIC '9' INIT(0);
    OPEN FILE(RF),FILE(SYSPRINT) LINESIZE(133);
    ON ENDFILE(RF) GOTO LAST;
RD: READ FILE(RF) INTO(RR);
    TCNT = TCNT + 1 ;
    IF ECD = '1' & ICD = ' ' & SEX = '1' THEN CALL TTL2 ;
    IF LCNT = 0 THEN CALL TTL1;
    IF ECD = '2' THEN ECD = 'E'; ELSE ECD = ' ';
    IF ICD = ' ' THEN ICD = 'TTT';
    IF SEX = '1' THEN DO; ICD = ' '; ECD = ' '; END;
        IF SEX = '1' THEN SEX = 'T';
    ELSE IF SEX = '2' THEN SEX = 'M';
    ELSE SEX = 'F';
    PUT SKIP(2) EDIT(ECD,ICD,SEX,(AGE(K) DO K = 1 TO 21))
        (A,A,X(1),A,X(1),21 F(6));
    LCNT = LCNT + 2;
    IF LCNT = 54 THEN DO ; PUT PAGE; LCNT = 0; END;
    GOTO RD;
TTL1: PROC;
    PUT SKIP EDIT(' ICD S', ' TOT', ' -1',
        ' -4', ' -9', ' -14', ' -19', ' -24',
        ' -29', ' -34', ' -39', ' -44', ' -49',
        ' -54', ' -59', ' -64', ' -69', ' -74',
        ' -79', ' -84', ' 85+', ' N/A')
        (22 A);
    PUT SKIP EDIT((133)'=')(A);
END TTL1;
TTL2: PROC;
    PUT PAGE;
    PUT SKIP EDIT((42)'*')(A);
    PUT SKIP EDIT('* THE AMOUNT OF DEATH BY SEXES AGE CAUSE *')
        (A);
    PUT SKIP EDIT((42)'*')(A);
    LCNT = 0;
END TTL2;
LAST:

```

SVD00010  
SVD00020  
SVD00030  
SVD00040  
SVD00050  
SVD00060  
SVD00070  
SVD00080  
SVD00090  
SVD00100  
SVD00110  
SVD00120  
SVD00130  
SVD00140  
SVD00150  
SVD00160  
SVD00170  
SVD00180  
SVD00190  
SVD00200  
SVD00210  
SVD00220  
SVD00230  
SVD00240  
SVD00250  
SVD00260  
SVD00270  
SVD00280  
SVD00290  
SVD00300  
SVD00310  
SVD00320  
SVD00330  
SVD00340  
SVD00350  
SVD00360  
SVD00370  
SVD00380  
SVD00390  
SVD00400  
SVD00410  
SVD00420  
SVD00430  
SVD00440  
SVD00450  
SVD00460  
SVD00470  
SVD00480  
SVD00490  
SVD00500  
SVD00510  
SVD00520  
SVD00530  
SVD00540  
SVD00550

FILE: SVDNT914 TAB999 A1 VM/SP RELEASE 2.1 EXPRESS PUT8302+ SLU203

```
PUT SKIP EDIT('INPUT COUNT  =',TCNT)(A,A);          SVD00560
PUT SKIP EDIT('SUMMR COUNT  =',OCNT)(A,A);          SVD00570
CLOSE FILE(RF),FILE(SYSPRINT);                      SVD00580
END GOLD;                                           SVD00590
/*                                                  SVD00600
//GO.RF      DD UNIT=TAPE,DSN=HH,VOL=SER=123321,LABEL=(2,BLP), SVD00610
//          DCB=(RECFM=FB,BLKSIZE=3960,LRECL=132),DISP=(OLD,KEEP) SVD00620
//GO.SYSOUT DD SYSOUT=A                             SVD00630
//GO.SYSPRINT DD SYSOUT=A                           SVD00640
//                                                  SVD00650
```

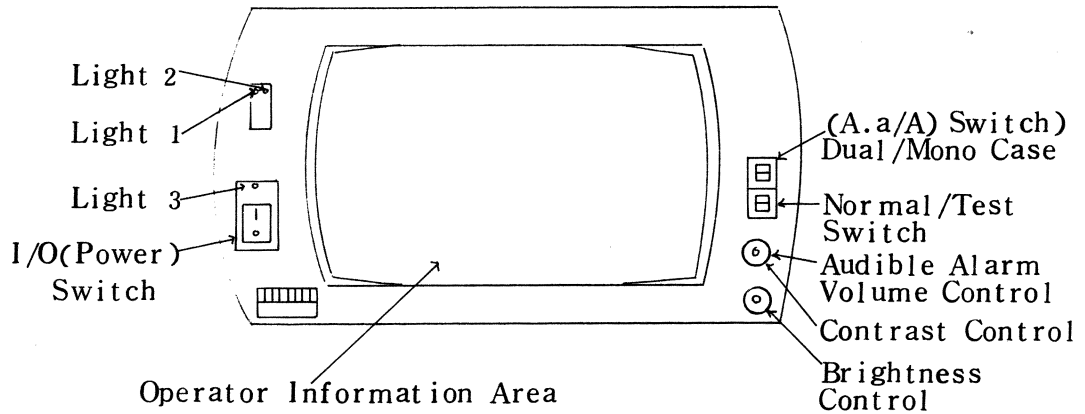
## 4. 人口動態申告CRT操作

### 方法 및 用語解説

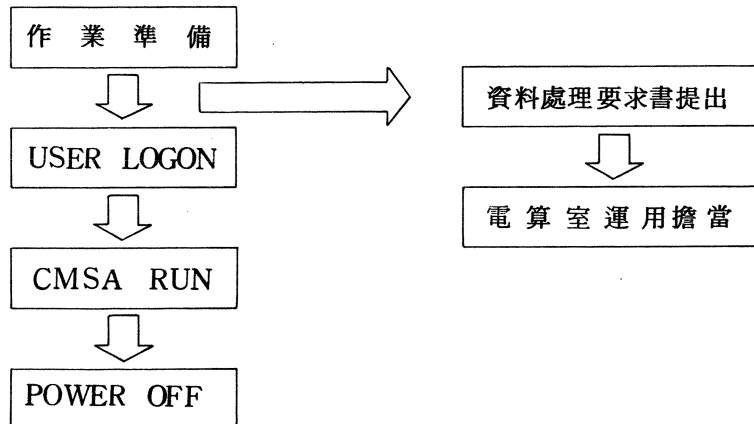
# 1. CRT 操作方法

## 가. 3278 CRT LIGHTS 및 SWITCHES

(1) 3278 畫面



(2)



## 나. 作業準備

- (1) 電源 PLUG 連結
- (2) NORMAL/TEST SWITCH 를 NORMAL 에 位置

(3) POWER SWITCH ON

- ■ 부분을 누른다.
- LIGHT 3에 불이 들어 온다.
- LIGHT 1에 불이 들어 온다.
- 잠시후 LIGHT 2에 불이 들어 온다.

(4) VM / 370 화면이 나타난다.

VM/370 ONLINE

```

          VV      VV      MM      MM
          VV      VV      MMM     MMM
          VV      VV      MMMM    MMMM
          VV      VV      MM MM   MM MM
333333333333 777777777777MMMM 00000000
333333333333 777777777777 MM 0000000000
33      VV33   77VV   77      00MM    00
          V33   VV    77M    00MM    00
          33    VV    77MM   00MM    00
3333VV  VV    77 MM   00MM    00
3333 VVVV   77 MM   00MM    00
          33 VV   77 MM   00MM    00
          33     77    00     00
33      33 VM/SP 77     00     00
333333333333 REL2.1 77 PUT8302+ 0000000000
333333333333 EXPRESS 77 SLU203 00000000
                                     RUNNING  BOSCENTR

```

(5) ENTER KEY를 누른다.

(6) VM / 370 ONLINE 화면이 사라지고 화면 右側下段에 CP PEAD가 나타난다.

다. USER LOGON

SYSTEM에 登錄된 USER는 다음과 같이 LOGON한다.

가령 USER-ID가 CMSA이고 PASS WORD가 CMSA이면

- (1) L CMSA CMSA을 KEY IN하고 ENTER KEY를 누르면 CRT 右側下段에 VM READ가 나타나고 다음과 같이 영상한다.

L CMSA CMSA  
 LOGON AT 11:39:43 KST MONDAY 05/21/84  
 VM/SP 2.1 8302+ SLU203 05/01/83

VM READ BOSCENTR

(2) B를 누르고 ENTER KEY를 누르면 다음과 같이 영상한다.

```

B
R: T=0.03/0.09 11:39:57
R: T=0.01/0.01 11:39:57
R: T=0.22/0.39 11:39:59
  
```

(3) FL를 KEY IN하고 ENTER KEY를 누르면 다음과 같은 화면이 나타난다.

```

LVL 0 --- A 191 3600 BLKS 3350 R/W 259 FILES 95% -- FILE 64 OF 259
SRVUSB14 SORTOLD A1 F 80 12 1 1/31/84 15:11
SRVWAC14 ACCEPT A1 F 80 88 7 4/23/84 10:53
SRVWEK14 ISAMEK A1 F 80 60 5 4/23/84 10:53
SRVWEK14 SRVBK A1 F 80 89 7 2/02/84 14:52
SRVWCC14 ISANCR A1 F 80 92 8 5/15/84 10:22
SRVWCR14 ISANCR A1 F 80 92 8 4/23/84 10:56
SRVWDC14 COBOL A1 F 80 221 18 4/24/84 14:55
SRVWED14 RVDET A1 F 80 414 33 4/24/84 14:54
SRVWEE14 SRVEDIT A1 F 80 698 55 5/15/84 10:27
SRVWET14 RVEDIT A1 F 80 699 55 5/18/84 11:37
SRVWGP14 LINK A1 F 80 712 56 4/23/84 10:55
SRVWGP14 PLIOPT A1 F 80 785 62 5/18/84 16:09
SRVWTA14 OLD A1 F 80 29 3 2/01/84 11:10
SRVWTA14 TAB234 A1 F 80 440 35 4/24/84 14:55
SRVWTB14 PLIOPT A1 F 80 130 11 4/23/84 10:54
SRVWTB14 TAB567 A1 F 80 429 34 4/27/84 10:00
SRVWTL14 PLIOPT A1 F 80 129 11 4/23/84 10:54
SRVWTT14 CORRECTI A1 F 80 61 5 2/20/84 13:11
SRVWTY14 OLD A1 F 80 36 3 2/01/84 11:14
SRVWTZ14 NEW A1 F 80 36 3 2/01/84 11:14
SRVWT114 TAB1 A1 F 80 162 13 4/27/84 10:00
SRVWUT14 RVUPDATE A1 F 80 324 26 4/24/84 14:54
1=HLP 2=BRW 3=END 4=XED 5=OS 6=/SB 7=SCB 8=SCF 9=/SD 10=/ST 11=>I 12=CAN
  
```

- (4) X를 KEY IN하고 ENTER KEY를 누르면 다의(3) 화면이 지워지고 다음과 같은 화면이 나타난다.

```

SRVWER14 SRVWR  A1  F 80  TRUNC=80  SIZE=89  LINE=0  COLUMN=1
====)
00000 * * * TOP OF FILE * * *
      1.....2.....3.....4.....5.....6.....7...
00001 //SRVWER14 JOB  TYPRUN=HOLD
00002 //BACKUP EXEC COBUCLG
00003 //COB.SYSIN DD *
00004         ID DIVISION.
00005         PROGRAM-ID.  RVBACK.
00006         ENVIRONMENT DIVISION.
00007         INPUT-OUTPUT SECTION.
00008         FILE-CONTROL.
00009             SELECT ISAM-F ASSIGN DA-I-DISK1
00010                 ACCESS MODE RANDOM
00011                 NOMINAL KEY NOMI-KEY
00012                 RECORD KEY JOSA-KEY.
00013             SELECT T-F  ASSIGN UT-S-TAPEOUT.
00014             SELECT T-A  ASSIGN UT-S-TAPEIN.
00015         DATA DIVISION.
00016         FILE SECTION.
00017         FD ISAM-F  BLOCK CONTAINS 16 RECORDS
00018                 LABEL RECORD STANDARD.
00019         01 ISAM-R.
PFKEY 1=D11 2=TOP 3=QQ 4=TAB 5=ROT 6=FILE 7=BA 8=FD 9=REP 10=SP 11=JO 12=U11

```

- (5) 화면 다의(4)인 상태에서 PROGRAM 및 DATA를 필요한 部分만 修正 補完 作業을 한다.
- (6) 修正作業이 끝나면 반드시 FILE를 KEY IN하고 ENTER KEY를 누른다.



```

SRVWBK14 SRVBK   A1  F 80  TRUNC=80 SIZE=89 LINE=0 COLUMN=1
==> FILE
00000 * * * TOP OF FILE * * *
      |...+...1...+...2...+...3...+...4...+...5...+...6...+...7...
00001 //SRVWBK14 JOB TYPRUN=HOLD
00002 //BACKUP EXEC COBUCLG
00003 //COB.SYSIN DD *
00004     ID DIVISION.
00005     PROGRAM-ID.  RVBACK.
00006     ENVIRONMENT DIVISION.
00007     INPUT-OUTPUT SECTION.
00008     FILE-CONTROL.
00009         SELECT ISAM-F ASSIGN DA-I-DISK1
00010             ACCESS MODE RANDOM
00011             NOMINAL KEY NOMI-KEY
00012             RECORD KEY JOSA-KEY.
00013         SELECT T-F   ASSIGN UT-S-TAPEOUT.
00014         SELECT T-A   ASSIGN UT-S-TAPEIN.
00015     DATA DIVISION.
00016     FILE SECTION.
00017     FD ISAM-F BLOCK CONTAINS 16 RECORDS
00018         LABEL RECORD STANDARD.
00019     01 ISAM-R.
PFKEY 1=D11 2=TOP 3=QQ 4=TAB 5=BOT 6=FILE 7=BA 8=FO 9=REP 10=SP 11=JO 12=U11

```

(7) 作業을 處理하기 위하여 PROGRAM을 電算室로 보내고자할시 SEND을 KEY IN하고 ENTER KEY를 누른다.

LVL	0	---	A	191	3600	BLKS	3350	R/W	259	FILES	95%	--	FILE	64	OF	259
SRVUSB14									F	80	12	1	1/31/84	15:11		
SRVWAC14									F	80	88	7	4/23/84	10:53		
SRVWBK14									F	80	60	5	4/23/84	10:53		
SRVWBK14									F	80	89	7	2/02/84	14:52		
SRVWCC14									F	80	92	8	5/15/84	10:22		
SRVWCR14									F	80	92	8	4/23/84	10:56		
SRVWDC14									F	80	221	18	4/24/84	14:55		
SRVWED14									F	80	414	33	4/24/84	14:54		

(8) PROGRAM을 PRINT하여 보기를 원할시 LPT 3를 KEY IN하고 ENTER KEY를 누른다.

LVL	---	A	191	3600	BLKS	3350	R/W	259	FILES	95%	---	FILE	64	OF	259
SRVUSB14									F	80		12	1	1/31/84	15:11
SRVWAC14									F	80		88	7	4/23/84	10:53
SRVWBK14									F	80		60	5	4/23/84	10:53
SRVWBK14									F	80		89	7	2/02/84	14:52
SRVWCC14									F	80		92	8	5/15/84	10:22
SRVWCR14									F	80		92	8	4/23/84	10:56
SRVWDC14									F	80		221	18	4/24/84	14:55
SRVWED14									F	80		414	33	4/24/84	14:54
SRVWEE14									F	80		698	55	5/15/84	10:27

- (9) 修正作業이 끝나면 QQ를 KEY IN하고 ENTER KEY를 누르면 나의 (7)번과 같은 화면이 나온다.

```

SRVWBK14 SRVRK   A1  F 80  TRUNC=80  SIZE=89  LINE=0  COLUMN=1
==== QQ
00000 * * * TOP OF FILE * * *
      1.....1.....+.....2.....+.....3.....+.....4.....+.....5.....+.....6.....+.....7...
00001 //SRVWBK14 JOB  TYPRUN=HOLD
00002 //BACKUP   EXEC COBUCLG
00003 //COB.SYSIN DD *
00004           ID  DIVISION.
00005           PROGRAM-ID.  RVBACK.
00006           ENVIRONMENT DIVISION.
00007           INPUT-OUTPUT SECTION.

```

- (10) /C를 KEY IN하고 ENTER KEY를 누르면 화면이 지워진다.

LVL	---	A	191	3600	BLKS	3350	R/W	259	FILES	95%	---	FILE	64	OF	259
SRVUSB14									F	80		12	1	1/31/84	15:11
SRVWAC14									F	80		88	7	4/23/84	10:53
SRVWBK14									F	80		60	5	4/23/84	10:53
SRVWBK14									F	80		89	7	2/02/84	14:52
SRVWCC14									F	80		92	8	5/15/84	10:22
SRVWCR14									F	80		92	8	4/23/84	10:56

- (11) LOGOFF를 KEY IN하고 ENTER KEY를 누르면 나의 (4)번과 같은 화면이 나오며 CRT상의 모든 作業이 끝남.

라. 用語解説

(1) FILE LISTING 상태의 PF KEY

PF1 = HLP : HELP INFORMATION, CMS 使用法 收錄

PF2 = BRW : BROWSE. X EDIT처럼 FILE의 內容을 修正할 수 없고  
EYE CHECK만 할 수 있으며 X EDIT보다 進行速度가 빠  
르다.

PF3 = END : FILE 修正後 終了時 使用 ( PF1, 2 와 同一 )

PF4 = X EDIT : FILE의 內容을 볼 때 使用

PF5 = OS : OS로 作業을 SEND할 때 使用

PF6 = /SB : FILE의 크기에 따라 SORT함 ( SORT BIG )

PF7 = SCB : SCREEN BACK ( 畫面의 BACK )

PF8 = SCF : SCREEN FORWARD

PF9 = /SD : SORT OF DATE ( FILE의 最近 FILE 順으로 SORT )

PF10 = /ST : SORT OF TYPE ( FILE TYPE 별로 SORT )

PF11 = > I : 使用않음.

PF12 = CAN : FILE CANCEL

(2) X EDIT 상태의 PF KEY

PF1 = D11 : DOWN 11 ( 위로 11 LINE 올라감 )

PF2 = TOP : FILE의 맨상단

PF3 = QQ : QUIT ( 修正事項이 없을 때 )

PF4 = TAB : TAB POSITION 命令後 FIELD KEY ( 좌우동작 )

PF5 = BOT : FILE의 맨하단

PF6 = FILE : 修正後 使用

PF7 = BA : BACK ( 지나간 畫面을 다시 보코자 할 때 )

PF8 = FO : FORWARD

PF9 = REP : 命令의 反復될 때 使用

PF10 = SP : 한 LINE의 內容을 分離하고자 할 때 使用

PF11 = JO : 2 LINE의 內容을 한 LINE으로 만들 때

PF12 = U11 : UP11 (아래로 11 LINE 내려감)

※ 參考事項

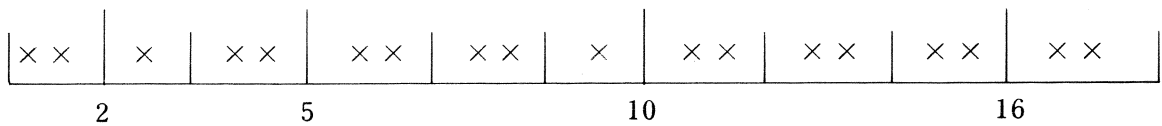
資料處理 教育用 ( 83 - 5 )

「SYSTEM 利用者 指針書」參照 바람.

## 5. 資料處理要求書作成方法

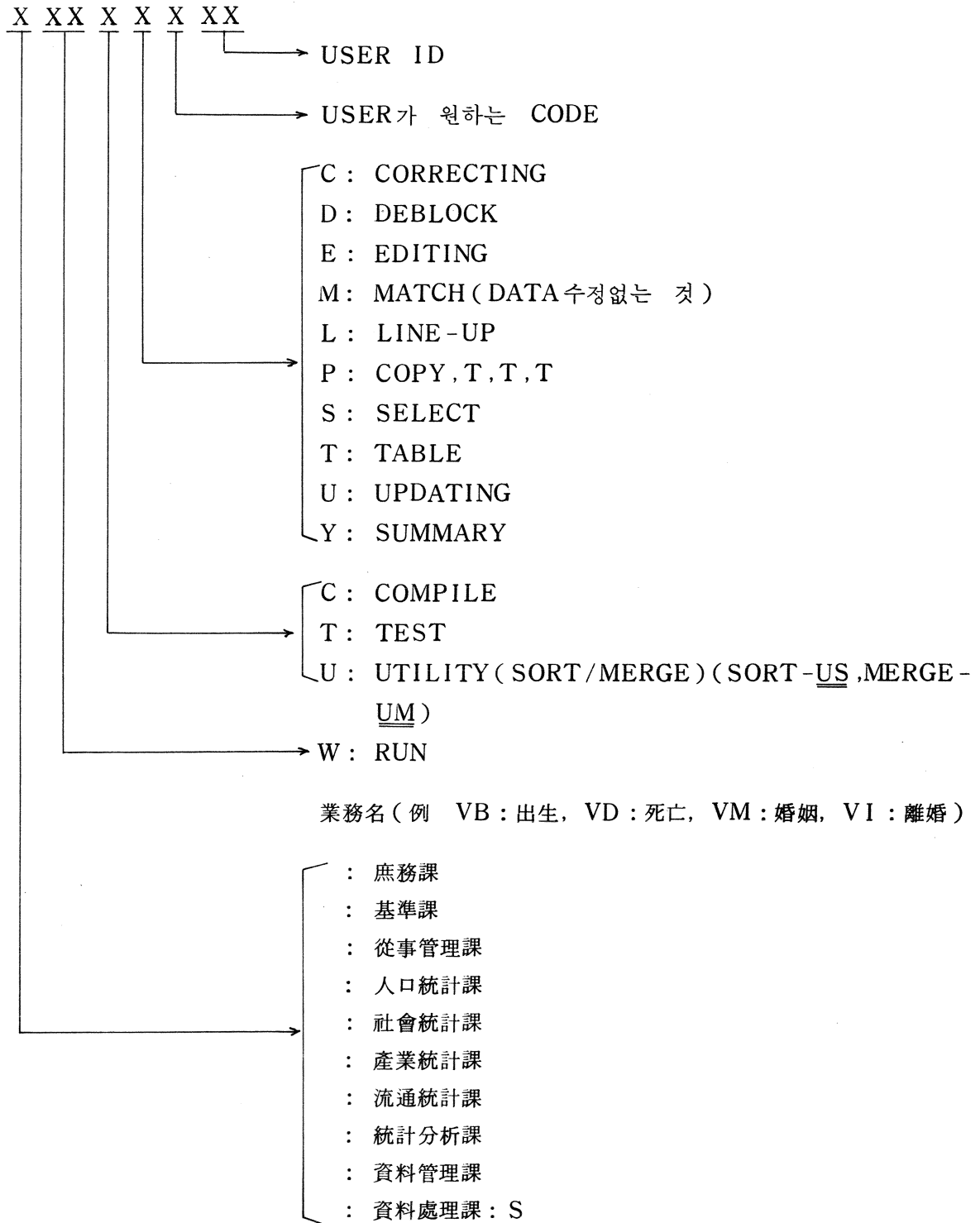
- 資料處理作業을 始作하기 前 다음과 같은 資料處理要求書에 該當事項을 記入하여 電子計算室 運用擔當者의 確認을 得한 後에 作業을 始作하여야 한다.
- 年度가 바뀔 때는 Program의 修正은 없으나 Data File 을 調整해야 하므로 반드시 System 關係者 或은 Program作成者에게 問議해야 한다.
- 作業을 完了後에는 Back-up File 을 保管해야 하므로 電算室에 作業이 끝났음을 알리고 File 保管 狀態를 確認할 것.
- Back-up Tape 作業時 label 은 다음 Code 形式에 따라 作成한다.

### 테 이 프 作 業 電 算 化 CODE



年度	課	業務名	過程	치수 01 ? 99	RUN TEST	USER CODE	도·월 分 期	USER 使用	REEL 수 2-1 2-2
			↓						
			CR	CORRECTION					
			DB	DEBLOCK					
			ED	EDITING OK					
			ER	EDITING ERROR					
			MT	MATCH					
			LP	LINE-UP					
			TT	TTT					
			TD	TTD					
			DT	DTT					
			ST	SELECT					
			UP	UPDATE					
			SM	SUM					
			SO	SORT					
			MG	MERGE					
			TB	TABLE					
			BK	BACK UP					
			MS	MASTER					
			CV	CONVERT					

PROGRAM作業 CODE標準化











# 자 료 처 리 요 구 서

개발담당	운영담당

업무명 : 과정명 : UPDATE 기재번호 : 10매중 4매

PROGRAM 명 : SVBWPB 14		(영문) 한글·영문		USER ID: CMSA		소요예정시간: 시간 5 분	
FILE		외 부 표 찰		I/O		D I T T O	
D/T	번호					C H RL. BL	
T		82 SVBDT00R1401ER		I		RL BL	
D		WK2222		I/O		RL BL	
		LIST		O		RL BL	
						RL BL	
						RL BL	
						RL BL	
						RL BL	
참고사항				P . G			
TAPE : 1월분 1 차수정				작성일자		월	일
				O . P		월	일
				작성일자		월	일

## 자 료 처 리 요 구 서

	개발담당	운영담당	
업무명 :	과정명 : EDIT	기재번호 : 10 매중 5 매	
PROGRAM 명 : SVBWET14	한글·영문	USER ID : CMSA	소요예정시간 : 시간 10 분
FILE	외 부 표 찰	I/O	D I T T O
D/T 번호			C H RL. BL
D	WK2222	I	RL BL
			RL BL
	LIST	O	RL BL
			RL BL
			RL BL
			RL BL
			RL BL
참고사항	P . G		
1월분 1차 EDIT 작업	작성일자	월 일	월 일
	O . P		
	작성일자	월 일	월 일

## 자 료 처 리 요 구 서

개발담당		운영담당	
업무명 :                      과정명 : ACCEPT                      기재번호 : 10 매 중 6 매                      소요예정시간 :                      시간 5 분			
PROGRAM 명 : SVBWAC14		USER ID : CMSA	
한글·영문		D I T T O	
F I L E	외 부 표 찰	I/O	C H
D/T 번호	Wk 2222	I/O	RL BL
D		V	RL BL
			RL BL
	LIST	O	RL BL
			RL BL
			RL BL
			RL BL
			RL BL
참고사항			
P . G			
	작성일자	월 일	월 일
	O . P		
	작성일자	월 일	월 일

## 자 료 처 리 요 구 서

	개발담당	운영담당	
업무명 :	과정명 : BACKUP	기재번호 : 10 매중 7 매	
PROGRAM 명 : SVWBK14		한글·영문	USER ID : CMSA      소요예정시간 : 시간 10 분
F I L E	외 부 표 찰	I/O	D I T T O
D/T 번 호			C/H    R L .    B L
D	WK 2222	I	보관여부 V
T	82 SVBBK00R1401	O	V
			RL BL
			RL BL
			RL BL
			RL BL
			RL BL
			RL BL
참고사항		P . G	
82년 1월 MASTER		작성일자	월 일      월 일
		O . P	
		작성일자	월 일      월 일

## 자 료 처 리 요 구 서

	개발담당	운영담당	
업무명 :	과정명 : MERGE	기재번호 : 10 매중 8 매	
PROGRAM 명 : SVBWMG14	한글·영문	USER ID : CMSA	소요예정시간 : 시간 30 분
F I L E	외 부 포 찰	I / O	D I T T O
D / T 번 호			C H   R L . B L
T R 1	82SVBBK00R1401	}	R L B L
:		I	R L B L
R 1 2	82SVBBK00R1412		R L B L
			R L B L
			R L B L
T	82SVBMGYRR14	O	R L B L
			R L B L
참고사항			
P . G			
	작성일자	월 일	월 일
	O . P		
	작성일자	월 일	월 일

## 자 료 처 리 요 구 서

	개발담당	운영담당	
업무명 : 과정명 : SORT 기재번호 : 10 메종 9 매 PROGRAM 명 : SVBUSB14 한글·영문 USER ID : CMSA 소요예정시간 : 시간 30 분			
FILE	외 부 표 찰	I/O	D I T T O
D/T	번호	번호	C H RL. BL
T	82SVBMGYRR14	I	RL BL
			RL BL
T	82SVBS0YRR14	O	RL BL
			RL BL
			RL BL
			RL BL
			RL BL
참고사항			
P . G		월 일	월 일
작성일자		월 일	월 일
O . P		월 일	월 일
작성일자		월 일	월 일





## 자 료 처 리 요 구 서

개발담당		운영담당	
업무명 : 인구동태신고    과경명 : SUMMARY    기재번호 : 3 매중 1 매    소요예정시간 : 1 시간    분			
PROGRAM명 : SVBWS114		USER ID: CMSA	분
F I L E		한글·영문	D I T T O
D/T	번호	I/O	C/H    RL.    BL
TR1	82 SVBMS00R14YR	I	보관여부 $\checkmark$
R2	82 SVBMS00R14YR	I	$\checkmark$
R3	82 SVMMS00R14YR	I	$\checkmark$
R4	82 SVIMS00R14YR	I	$\checkmark$
T	82 SVBSM01R14	O	$\checkmark$
참고사항		P . G	월 일
		작성일자	월 일
		O . P	월 일
		작성일자	월 일





## 6. 人口動態結果表

제 1 표

## 발생년도 및 신고년도별 동태건수

발생 신고 년도	발 생 년 도											
	총 계	1982	1981	1980	1979	1978	1977	1976	1975	1974	1973	1972
출 생												
1982												
1981												
⋮												
1972												
사 망												
1982												
1981												
⋮												

제 2 표

## 발생지역 및 발생년도별 동태건수

발생 지역	발 생 년 도											
	총 계	1982	1981	1980	1979	1978	1977	1976	1975	1974	1973	1972
전 국												
출 생												
사 망												
혼 인												
이 혼												
서 울												
출 생												
⋮												
종로구												
출 생												
⋮												

제 3 표

## 월 별 출 생

월 별	지 역														
	전국	서울	부산	대구	인천	경기	강원	충북	충남	전북	전남	경북	경남	제주	기타
1982 년															
남															
여															
1 월															
남															
여															
2 월															
남															
여															
∴															
12 월															
남															
여															

제 4 표

## 모의연령 및 출산순위별 출생

모 의 연 령	출 산 순 위														
	총 계	1아	2아	3아	4아	5아	6아	7아	8아	9아	10아	11아	미상		
	계	남	여	남	여	남	여	남	여	남	여	남	여	남	여
1982 년															
10~14															
10															
11															
12															
13															
14															
15~19															
15															
16															
17															
18															
19															
20~24															
∴															

제 5 표

모의연령 및 출산순위별(지역별) 출생

모의연령 및 지 역	출 산 순 위													
	총 계	1 아	2 아	3 아	4 아	5 아	6 아	7 아	8 아	9 아	10 아	11 아	미 상	
	계 남 여	남 여	남 여	남 여	남 여	남 여	남 여	남 여	남 여	남 여	남 여	남 여	남 여	
1982 년														
10~14														
15~19														
⋮														
50 +														
미 상														
서울														
10~14														
15~19														
⋮														

제 6 표

모의연령, 교육정도 및 지역별 출생

모의연령 및 교육정도	지 역														
	전국	서울	부산	대구	인천	경기	강원	충북	충남	전북	전남	경북	경남	제주	기타
1982 년															
10~14															
15~19															
⋮															
50 +															
미 상															
미 취 학															
10~14															
15~19															
⋮															
50 +															
미 상															
국민학교															
⋮															



제 7 표

## 월 별 사 망

월 별	지 역														
	전국	서울	부산	대구	인천	경기	강원	충북	충남	전북	전남	경북	경남	제주	기타
1982년 남여															
1월 남여															
...															
12월 남여															
1981년 남여															
...															

제 8 표

## 연령 및 지역별 사망

연 령	지 역 별														
	전국	서울	부산	대구	인천	경기	강원	충북	충남	전북	전남	경북	경남	제주	기타
	계	남여	남여	남여	남여	남여	남여	남여	남여	남여	남여	남여	남여	남여	남여
1982년															
0~4															
0															
1															
2															
3															
4															
5~9															
5															
6															
7															
8															
9															
10~14															
10															
...															
95+															
연령 이상															

제 9 표

연령 및 직업별 사망

연령	총계	전문	행정	사무	판매직	서어비	농림	생산	무직	군인	미상
		기술직	관리직	및 관리직		스직	수산업	관련직			
1982년											
0											
1~4											
5~9											
10~14											
⋮											
95+											
이상											
남자											
0											
1~4											
⋮											

제 10 표

월 별 혼 인

년 월	지 역														
	전국	서울	부산	대구	인천	경기	강원	충북	충남	전북	전남	경북	경남	제주	기타
1982년															
1월															
2월															
⋮															
12월															
1981년															
1월															
2월															
⋮															

제 11 표

부처상호간 연령별 혼인

년 도의 연령	처 의 연 령								
	총 계	15세미만	15	16	.....	45~49	50~54	55 +	N/A
1982 년									
15세미만									
15 ~ 19									
20									
21									
22									
23									
24									
25									
⋮									
50 ~ 54									
55 +									
미 상									
1981 년									
⋮									
⋮									

제 12 표

연령 및 지역별 혼인

연 령	지 역															
	전국	서울	부산	대구	인천	경기	강원	충북	충남	전북	전남	경북	경남	제주	기타	
	계	남	여	남	여	남	여	남	여	남	여	남	여	남	여	
1982 년																
15세미만																
15																
16																
17																
⋮																
⋮																
76+																
N/A																
1981 년																
⋮																
⋮																

제 13 표

연도 및 혼인종류별 혼인

거주지 및 부 혼인회수	혼인년도 및 처의 혼인회수																			
	1982			1981			1980			.....	1975			1974			1973			
	계	초혼	재혼	미상	계	초혼	재혼	미상	계	초혼	재혼	미상	계	초혼	재혼	미상	계	초혼	재혼	미상
전국 계																				
초혼																				
재혼																				
미상																				
서울 계																				
초혼																				
재혼																				
미상																				
⋮																				
기타 계																				
초혼																				
⋮																				

제 14 표

월 별 이 혼

년 월	지 역														
	전국	서울	부산	대구	인천	경기	강원	충북	충남	전북	전남	경북	경남	제주	기타
1982															
1월															
2월															
3월															
⋮															
12월															
1981															
1월															
2월															
3월															
⋮															

제 15 표

부처상호간 연령별 이혼

부의연령	처 의 연 령												
	총계	15세미만	15~19	20	21	22	.....	55~69	60~64	65~69	70~74	75+	이 상
1982 년													
15세미만													
15~19													
20~24													
⋮													
65~69													
70~74													
75+													
N/A													
1981 년													
⋮													

제 16 표

연령 및 지역별 이혼

연 령	지 역															
	전국	서울	부산	대구	인천	경기	강원	충북	충남	전북	전남	경북	경남	제주	기타	
	남여	남여	남여	남여	남여	남여	남여	남여	남여	남여	남여	남여	남여	남여	남여	
1982 년																
15세미만																
16																
17																
⋮																
74																
75																
76+																
미 상																
1981 년																
⋮																

제 17 표

지역별 이혼사유별 이혼

이혼년도 사유	지 역														
	전국	서울	부산	대구	인천	경기	강원	충북	충남	전북	전남	경북	경남	제주	기타
1982															
부부불화															
가족간 불화															
건강상															
경제문제															
기타															
미상															
1981															
⋮															
⋮															
⋮															

총괄표(가)

81년도 발생 신고 사망자수

신고년도	신고건수	사인분류가능건수		사인미기재 및 불명확한 건수
			의사진단	
합계				
1981				
1982				

총괄표(나)

사망장소별 사망자수

사망장소	사망자수	백분비 (%)
계		
자택		
병원		
기타		

총괄표(다)

사망진단자별 사망자수

구분	총수	의사	한의사	기타
사망신고건수				
사인분류가능건수				

총괄표(라)

신고사망자의 성·연령별 구성( 전체 )

연 령	신 고 사 망 자		사 인 분 류 가 능		의 사 진 단	
		%		%		%
총 수						
0						
1~4						
5~9						
⋮						

총괄표(라)

신고사망자의 성·연령별 구성( 남자 )

연 령	신 고 사 망 자		사 인 분 류 가 능		의 사 진 단	
		%		%		%
총 수						
0						
1~4						
5~9						
⋮						



총괄표(라)

신고사망자의 성·연령별 구성(여자)

연령	신고사망자		사인분류가능		의사진단	
		%		%		%
총수						
0						
1~4						
5~9						
⋮						

총괄표(마)

신고사망자의 지역별 구성

지역	신고사망자수		사인분류가능		의사진단	
		%		%		%
전국						
시부						
군부						
서울						
부산						
⋮						
제주						
기타						

표 1-1

17 대분류에 의한 성·연령별 사망자수 (의사진단자료)

표준분류 번호	사망원인	총수	0	1~4	5~14	15~ 24	25~ 34	35~ 44	45~ 54	55~ 64	65~ 74	75+
001~999	사망의 남 모든원인 여											
001~139	감염성 남 기생충성 여											
140~238	⋮											
⋮	⋮											

표 1-2

17 대분류에 의한 성·연령별 사망자수 (분류가능자료)

표준분류 번호	사망원인	총수	0	1~4	5~14	15~ 24	25~ 34	35~ 44	45~ 54	55~ 64	65~ 74	75+
001~009												
001~139												
140~139												
⋮												

표 2-1

55 항목 특수분류에 의한 성·연령별 사망자수 (의사진단자료)

표준분류 번호	사망원인	총수	0	1~4	5~14	15~ 24	25~ 34	35~ 44	45~ 54	55~ 64	65~ 74	75+
001~999	사망의 모 든원인	계 남 여										
001~139	감염성· 기생충성	계 남 여										
001~009	장관감염명	계 남 여										
⋮	⋮											

표 2-2

55 항목 특수분류에 의한 성·연령별 사망자수 (분류가능자료)

표준분류 번호	사망원인	총수	0	1~4	5~14	15~ 24	25~ 34	35~ 44	45~ 54	55~ 64	65~ 74	75+
001~009	사망의 모 든원인	계 남 여										
001~139	⋮											
001~009												
⋮												

표 3-1

55 항목 특수분류에 의한 지역별 사망자수 (의사진단자료)

표준분류 번호	사망원인	전국		시부		대도시		중·소도시		군부	
		전수	구성비	전수	구성비	전수	구성비	전수	구성비	전수	구성비
001~999	사망의 모든원인										
001~139	감염성·기생충성										
001~009	⋮										
010~018	⋮										
⋮	⋮										

표 3-2

55 항목 특수분류에 의한 지역별 사망자수 (분류가능자료)

표준분류 번호	사망원인	전국		시부		대도시		중·소도시		군부	
		전수	구성비	전수	구성비	전수	구성비	전수	구성비	전수	구성비
001~999	사망의 모든원인										
001~139	감염성·기생충성										
001~009	⋮										
010~018	⋮										
⋮	⋮										

표 4 기본분류( 307 항목 )에 의한 성·연령별 사망자수( 의사진단자료 )

i.C.D	사 망 원 인	총수	0	1~4	5~9	10~14	...	15~19	20~24	...	85+
001~999	사망의모든원인 계										
	남										
	여										
001~009	계										
.....	.....										
.....	남										
.....	여										

표 5 성·연령별 사망원인 순위( 의사진단자료 )

연령·성	1 위		2 위		3 위		.....	9 위		10 위	
	사 인	구성비	사 인	구성비	사 인	구성비		사 인	구성비	사 망	구성비
전 체											
계											
남											
여											
0											
계											
남											
여											
1~4											
계											
남											
여											
.....											
.....											