

資料處理教材  
86-3

# 家 口 標 本 管 理

(電算處理開發報告書)

1986. 5.

經濟企劃院 調查統計局

이 報告書는 家口標本管理를 電算處理時 作業에 必要  
한 CRT 및 CICS 操作에 의한 DATA 修正 補完  
및 結果表算出 등을 누구나 쉽게 應用할 수 있도록 만  
든 冊입니다.

作業處理時에 問議事項이 있으시면 資料處理課로 連絡  
바랍니다.

資料處理課長            金 敬 福

(304) 金 滢 權

(306) 姜 桂 和

## 目 次

<b>1. SYSTEM FLOWCHART</b> .....	5
1) 가구표 .....	7
2) 변동보고서 .....	8
3) 표본조사표 .....	9
4) MASTER CREATION .....	10
5) 조사구교체 .....	11
6) 눈기별, 연별 TABLE 작업 .....	13
7) MASTER 재조정 작업 .....	14
<b>2. 入力 DESIGN</b> .....	15
1) 가구표 .....	17
2) 표본조사표 .....	18
3) 변동보고서 .....	19
4) 가구 MASTER .....	21
<b>3. 기계내검요령서</b> .....	23
1) 가구표 .....	26
2) 변동보고서 .....	34
3) 표본조사표 .....	44
<b>4. CRT (CATHODE RAY TUBE) 조작방법</b> .....	51
<b>5. CICS (CUSTOMER INFORMATION CONTROL SYSTEM) 사용방법</b> ...	59
1) 가구관리 MASTER ON-LINE SYSTEM FLOW .....	61

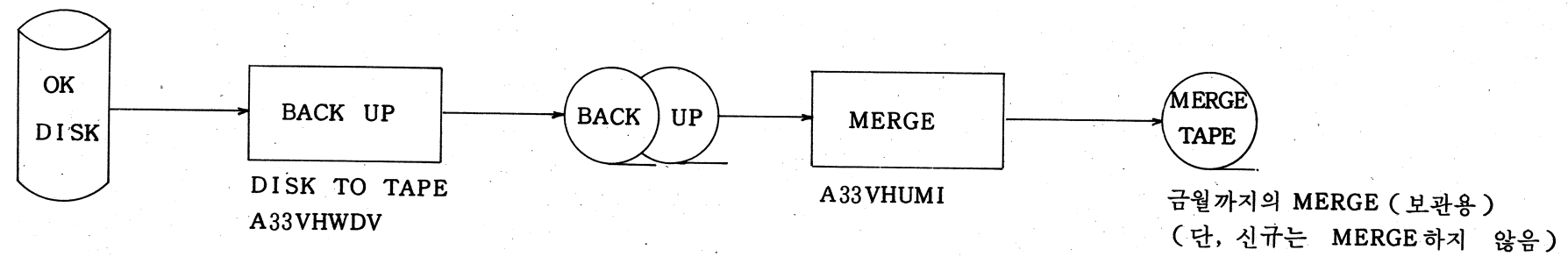
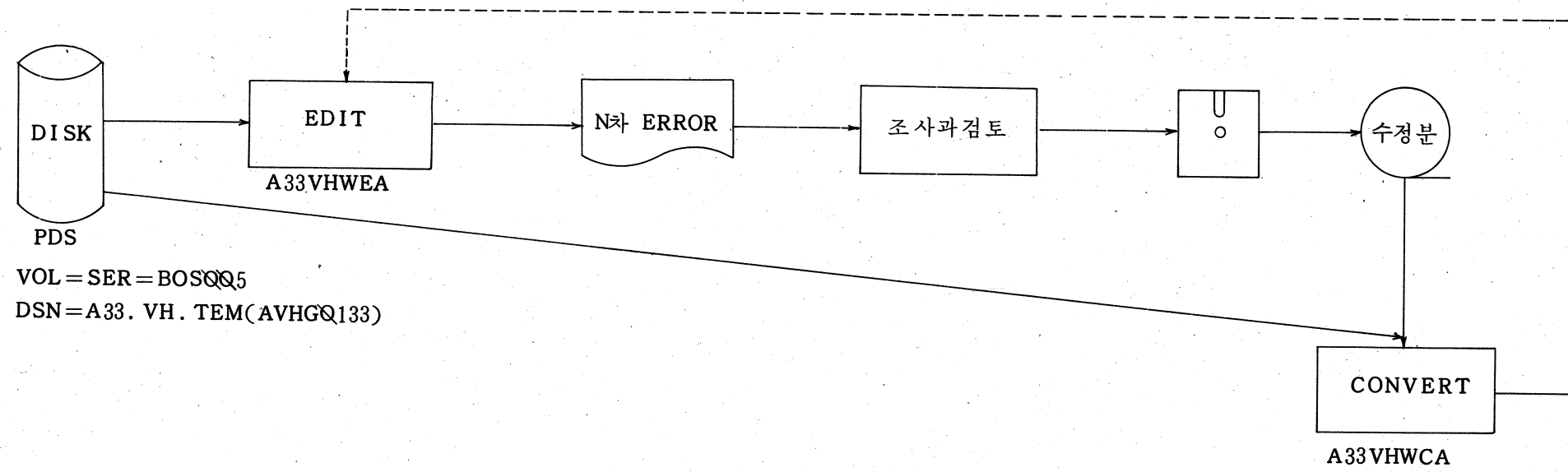
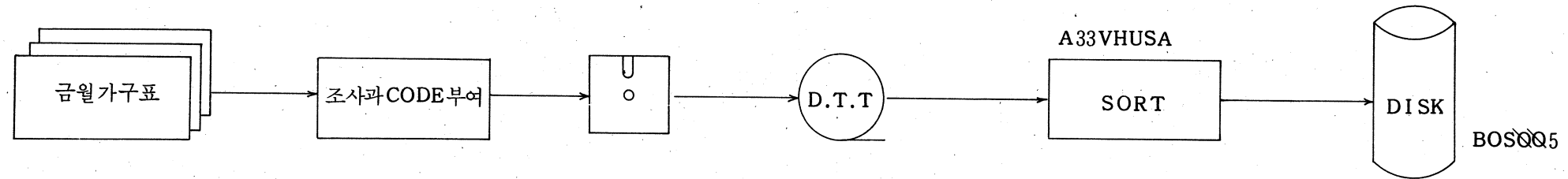
<b>6 . SOURCE PROGRAM LIST</b> .....	77
1) A33VHUSA : 금월가구표 SORT .....	79
2) A33VHWEA : 금월가구표 EDIT .....	80
3) A33VHWCA : 금월가구표 CONVERT .....	89
4) A33VHWDV : 금월가구 BACK - UP .....	90
5) A33VHUM1 : 금월가구 MERGE .....	91
6) A33VHUSB : 금월변동보고서 SORT .....	92
7) A33VHWEB : 금월변동보고서 EDIT .....	93
8) A33VHWBB : 금월변동보고서 CONVERT .....	100
9) A33VHUM2 : 금월변동보고서 MERGE .....	101
10) A33VHWT8 : TABLE .....	102
11) A33VHUSC : 금월표본조사표 SORT .....	104
12) A33VHWEP : 금월표본조사표 EDIT .....	105
13) A33VHWCP : 금월표본조사표 CONVERT .....	110
14) A33VHUM3 : 금월표본조사표 MERGE .....	111
15) A33VHTTA : 금월표본조사표 TABLE .....	112
16) A33VHWM6 : MASTER와 6 CARD MATCH .....	115
17) A33VHWM5 : MASTER와 5 CARD MATCH .....	119
18) A33VHINS : 가구표 INSERT .....	121
19) A33VHWMB: 1, 2, 3, 4, 7 CARD 변동 MATCH .....	125
20) A33VHWT5 : MASTER TABLE .....	128
21) A33VHMNB : MASTER 조사구 TABLE .....	130
22) A33VHWVV : MASTER BACK-UP .....	132



23) A 33V HWM1 : 신규가구표 INSERT .....	133
24) A 33V HWM2 : 교체구역 CHECK .....	137
25) A 33V HUSE : 조사구교체내역표 SORT .....	139
26) A 33V HWVH : 분기별 연별 조사표 TABLE .....	140
27) A 33V HWT2 : 분기별 연별 MASTER .....	151
28) A 33V HMAP : CICS MAP DESIGN .....	162
29) A 33V HPLI : CICS APPLICATION PROGRAM .....	166
30) A 33V HWVT : MASTER 재수정 .....	174
31) A 33V HUST : MASTER SORT .....	178
32) A 33V HWCR : MASTER 재 CREATION .....	179
33) A 33V HTDL : MASTER 잠정적 삭제 .....	180
34) A 33V HDEL : 분기별 삭제 .....	181
35) A 33V HWDI : MASTER 수정작업 ( SAM FILE 시 ) .....	182
36) A 33V HCRT : VSAM CREATION .....	184
37) A 33V HDIR : VSAM DIRECT UPDATE .....	188
38) A 33V HTAB : VSAM SEQUENTIAL UPDATE .....	189
<b>7. 결과표 양식</b> .....	<b>191</b>
<b>8. 조사표</b> .....	<b>219</b>
<b>9. TAPE 표찰방법</b> .....	<b>229</b>

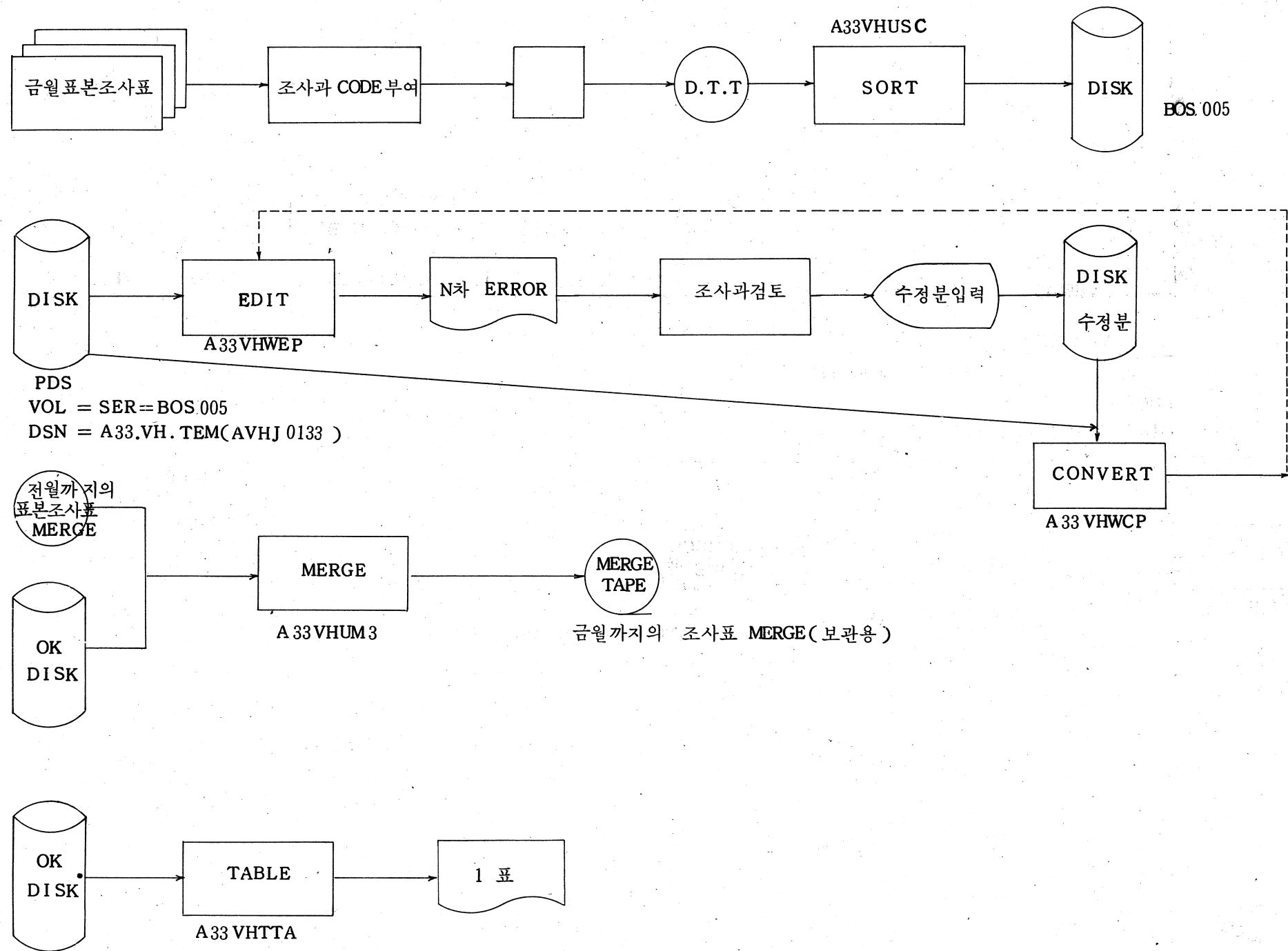
# 1. SYSTEM FLOWCHART

# 1. 가구표 및 신규가구표 FLOWCHART

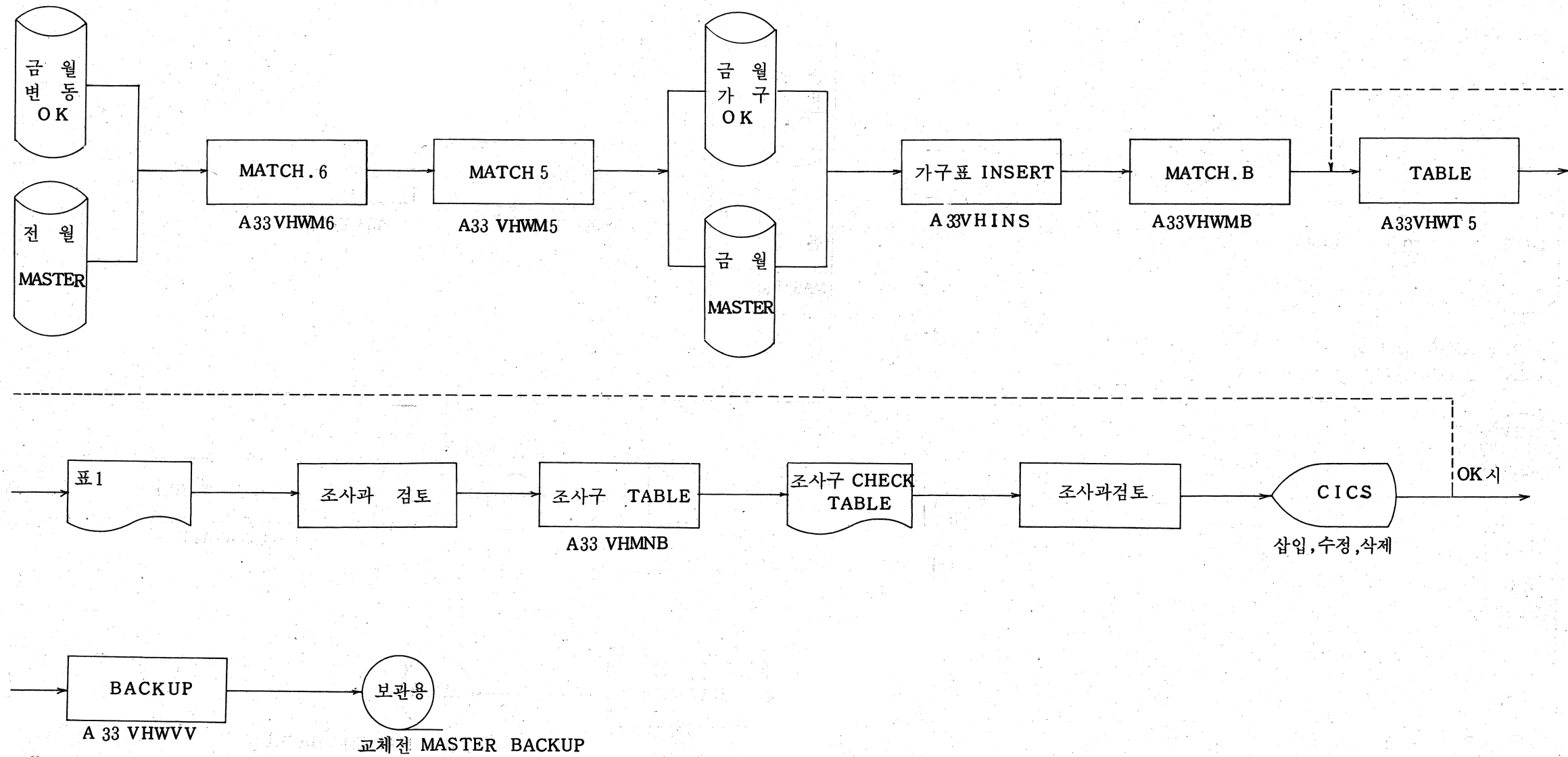




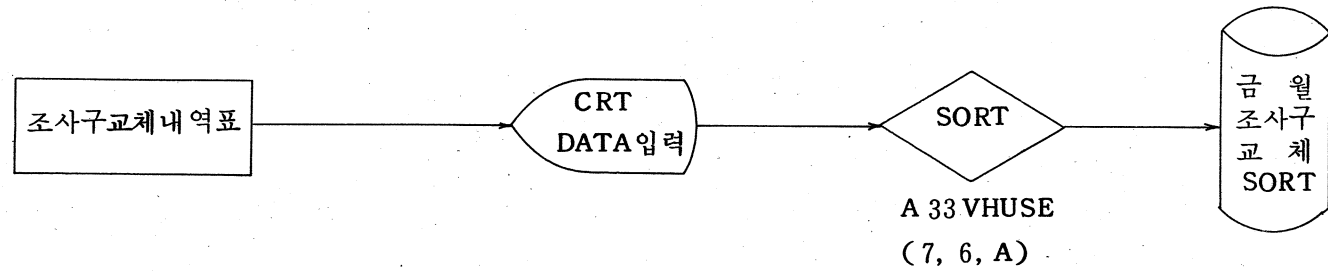
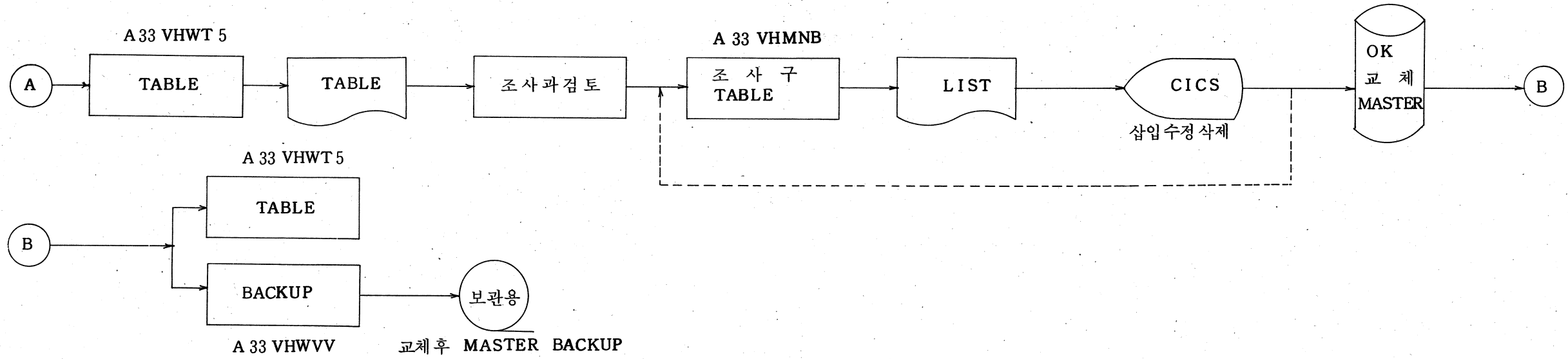
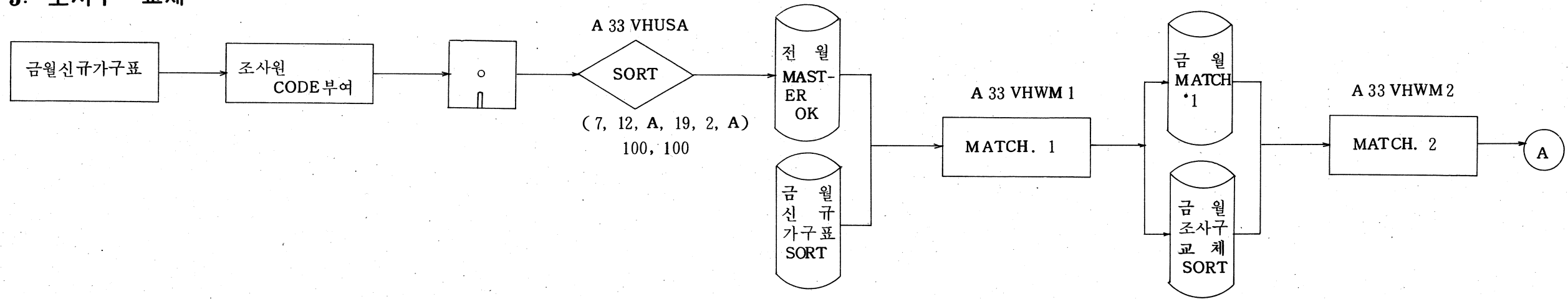
### 3. 표본조사표 FLOWCHART



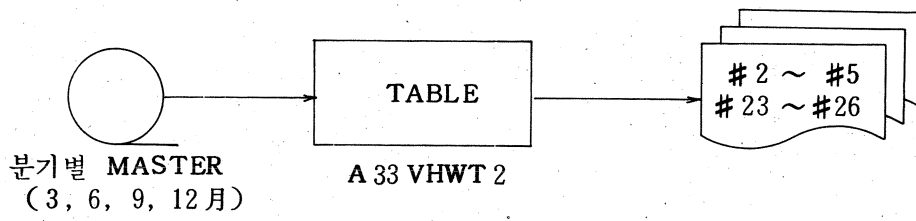
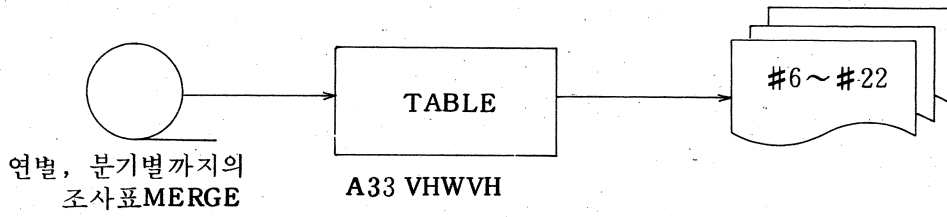
#### 4. MASTER CREATION



5. 조사구 교체

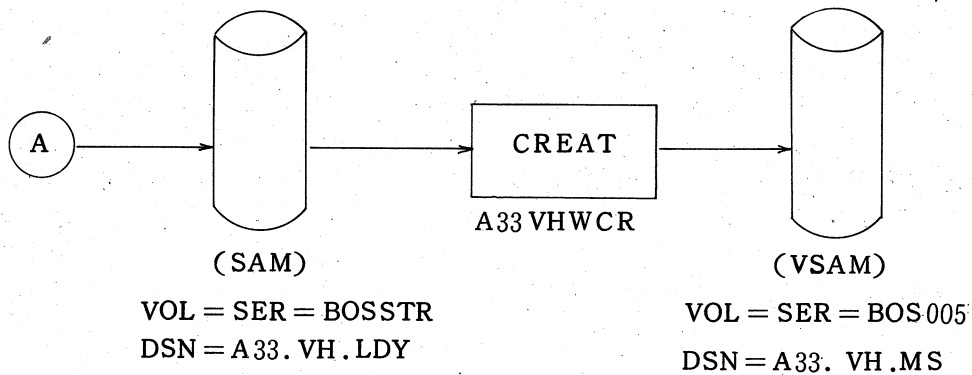
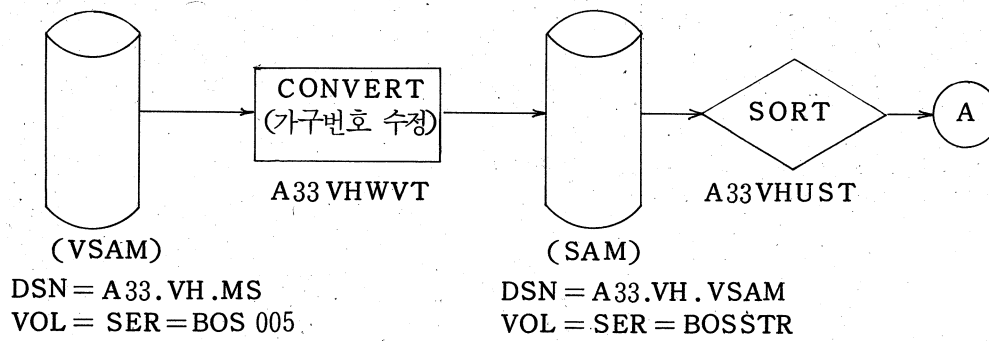


## 6. 분기별·연별 TABLE 작업





## 7. MASTER 재조정작업



## 2. 入 力 DESIGN

## MULTIPLE LAYOUT FORM

PROGRAM 인구동태표본 VOL LBL 가구표 FILE LBL RL 100 BF 40 BS 4,000 MFS NAME DATE

가 구 사 항	1	K E Y				가 구 주 소														
	일련번호	조사구번호	구역번호	거처번호	가구번호	가구번호	가구번호	가구번호	부속구	가구	가구	가구	가구							
	5	10	15	20	25	30	35	40	45	50	55	60	65	70	75	80	85	90	95	100
가 구 원 사 항	"	K E Y				가 구 주 소		성	생년월일	학 배	초혼년월	출산 계획	임신 여부	전입일자						
	"	"	"	"	"	가구원번호	성	년 월 일	학 배	초혼년월	출산 계획	임신 여부	전입일자	년 월 일	년 월 일	년 월 일	년 월 일	년 월 일	년 월 일	년 월 일
	5	10	15	20	25	30	35	40	45	50	55	60	65	70	75	80	85	90	95	100

## MULTIPLE LAYOUT FORM

PROGRAM 인구동태표본      VOL LBL 표본조사표      FILE LBL      RL 100      BF 40      BS 4,000      MFS      NAME      DATE

출생	일련번호	조사구번호	가구번호	구분	모에 관한 사항	출생아에 관한 사항	출생성	출생년월일	출생순위	임신기간	모의교육정도	조사자
	3			1	성명	성명	성	년월일				
5 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80 85 90 95 100												
사망				2	사망자이름	사망년월일	직업	사망년월일	사망원인	사망원인		
5 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80 85 90 95 100												
혼인				3	夫에 관한 사항	妻에 관한 사항	연령	교육정도	연령	교육정도	혼인년월	
5 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80 85 90 95 100												
이혼				4	夫에 관한 사항	妻에 관한 사항	연령	연령	생존자녀수	혼인년월	발생년월	
5 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80 85 90 95 100												
5 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80 85 90 95 100												

### MULTIPLE LAYOUT FORM

PROGRAM 인구동태표본      VOL LBL 변동보고서      FILE LBL      RL 100      BF 40      BS 4,000      MFS      NAME      DATE

가 구 증 가	2	일련번호	변동년월	조사구번호	카드구분1	가 구 번호	가 구 원 수	변동일자 년 월 일	전 거 주 지	이 동 사 유	구 분																				
												5 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80 85 90 95 100																			
가 구 감 소	"	"	"	"	2	"	가 구 원 수	변동일자 년 월 일	목 적 지	이 동 사 유	구 분																				
												5 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80 85 90 95 100																			
가 구 원 증 가	"	"	"	"	3	"	가 구 원 번호	성 명	가 구 주 관 계	생 년 월 일 년 월 일	학 령	배 우 관 계	초혼년월 년 월	출산 남 여	임신여부 년 월	결 과	전 거 주 지	전입일자 년 월 일	이 동 사 유	구 분											
																						5 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80 85 90 95 100									
가 구 원 감 소	"	"	"	"	4	"	가 구 원 번호	성 명	변동일자 년 월 일	목 적 지	이 동 사 유	구 분																			
													5 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80 85 90 95 100																		
가 구 표 수 정 요 구 사 항	"	"	"	"	5	"	가 구 원 번호	가 구 표 관 번호	수 정 전 내 용	수 정 후 내 용																					
											5 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80 85 90 95 100																				

### MULTIPLE LAYOUT FORM

PROGRAM	인구동태표본	VOL LBL	변동보고서	FILE LBL	RL	100	BF	40	BS	4,000	MFS	NAME	DATE												
조사구내이동	2		6		이	동	전	이	동	후	가	구	주	관	계	가	소	배	초	혼	년	월			
					가	구	원	가	구	원	가	구	원	가	구	원	가	구	원	가	구	원	가	구	원
임신기록및결과	"		7		가	구	원	성	명	출	산	에	정	결	년	월	과								

# MULTIPLE LAYOUT FORM

PROGRAM 인구등록표본      VOL LBL 가구 MASTER      FILE LBL      RL 100      BF 200      BS 20,000      MFS      NAME      DATE

가 구 사 항	1	K E Y				거	소	가	가	행													비	사	교											
		조	구	거	가	유	관	구	구	적	시	구	동	주 소											무	년	코	년	코	년	코	년	코			
		사	구	치	구	치	관	구	구	적	시	구	동												소	월	드	월	드	월	드	월	드			
가 구 원 사 항	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"	"
교 체 내 역 표	교	사	조	구	변											6 → 계속 (C) 7 → 신규 (S) 8 → 중지 (D)																				
	체	무	사	구	경											6 → 계속 (C) 7 → 신규 (S) 8 → 중지 (D)																				
	년	소	구	호	코																															
	월		번호	드	드																															

### 3. 기 계 내 검 요 령 서



기 계 내 용 검 사 요 령  
( DATA CHECK 요 령 )

## 기계 내용 검사 요령

CHECK 항목	정 CODE	COLUMN 위치		ERROR 기 호
		최 초	최 후	
<b>1. 가구표</b>				
(가 구 사 항)		1	66	
카드 번호	1	1	1	*
조사구 번호	662 개조사구	7	10	*****
구역 번호	01 ~ 24	11	12	**
거 처 번호	01 ~ 99	13	14	**
가 구 번호	0101 ~ 3030	15	18	*****
가 구 원 번호	00	19	20	NO GAGU SAHANG
거 처 구 분	1 ~ 5	21	21	*
소 유 구 분	1 ~ 6	22	22	*
구 역 구 분	1 ~ 2	23	23	*
가 구 구 분	1 ~ 4	24	24	*
가 계 조 사 부 적 격 사 유	01 ~ 15	25	26	**
주 소	한글	27	66	한글 ERR

(DATA CHECK 요령)

업무명

년 월 일

작성자

ERROR CODE 처리	비 고
<p>1이 아니면 ERROR</p> <p>662 개의 조사구가 아닐때</p> <p>구역번호가 01 ~ 24 가 아닐때</p> <p>거처번호가 01 ~ 99 가 아닐때</p> <p>가구번호가 0101 ~ 3030 이 아닐때</p> <p>가구사항 카드가 나타나지 않았을때</p> <p>거처구분이 1 ~ 5 가 아니었을때</p> <p>소유구분이 1 ~ 6 이 아니었을때</p> <p>구역구분이 1 ~ 2 가 아니었을때</p> <p>가구구분이 1 ~ 4 가 아닐때, 구역구분이 2 이고 가구구분이 1 ~ 4 일때</p> <p>가구구분이 4 이고 사유가 01 ~ 15 가 아닐때</p>	

CHECK 항목	정 CODE	COLUMN 위치		ERROR 기 호
		최 초	최 후	
(가 구 원 사 항)		1	73	
가 구 원 번 호	01 ~ 99	19	20	**
성 명	한 글	21	40	
가 구 주 관 계	0000 ~ 0940, 9999, 9988	41	44	****
성 별	1 ~ 2 , 9	45	45	*
생 년 월 일				*****
년	880 ~ 985	46	48	***
월	01 ~ 12	49	50	**
일	01 ~ 31	51	52	**

ERROR CODE 처리	비 고
<p>가구원번호가 01~99가 아닐때</p> <p>한글 ERROR 일때</p> <ol style="list-style-type: none"> <li>1. 가구주관계와 생년월일을 비교하여 가구주관계란 이 03 (미혼자녀) 또는 04 (미혼형제)일 경우 각 가구원 (형제자매)의 연령차이가 10개월 이상 아닐때</li> <li>2. 가구주와의 관계가 05 (부모) 및 조부모(06)인 경우 배우관계가 1. 유배우, 2. 사별, 3. 이혼이 아닐때</li> <li>3. 가구주관계가 0000 ~ 0940, 9999, 9988 이 아닐때</li> <li>4. 가구주와의 관계가 03 또는 04인 경우 배우 관계가 미혼이 아닐때 <ol style="list-style-type: none"> <li>1. 성별이 1~2 또는 9가 아닐때</li> <li>2. 남자(1)이면서 임신여부란이 공란이 아닐때</li> </ol> </li> </ol> <p>생년월일과 초혼년월을 비교하여 15세 미만일때</p> <p>생년이 880~985가 아닐때</p> <p>월이 01 ~ 12 가 아닐때</p> <p>일이 01 ~ 31 가 아닐때</p>	

CHECK 항목	정 CODE	COLUMN 위치		ERROR 기 호
		최 초	최 후	
학 력	000 ~ 183	53	55	***
배 우 관 계	1 ~ 4 , 9	56	56	*
초 혼 년 월	< 현 년 도	57	60	****
출 산 계 획	' , '00' , 22, 99	61	62	**
임 신 여 부		63	67	*
년	000 ~ 985			
월	00 ~ 12			
결 과	0 ~ 4			

ERROR CODE 처리	비 고
<p>1. 학력이 000 ~ 183 이 아닐때</p> <p>2. 생년월일과 학력을 비교하여 다음 범위에 속하지 않을때.</p> <ul style="list-style-type: none"> <li>○ 국민재 만 6세 ~ 만 14세 - 국민졸업중퇴 : 만 7세이상</li> <li>○ 중학재 만 12세 ~ 만 17세 - 중학졸업중퇴 : 만 13세이상</li> <li>○ 고등재 만 15세 ~ 만 20세 - 고등졸업중퇴 : 만 16세이상</li> <li>○ 대학재 만 18세이상 - 대학졸업중퇴 : 만 19세이상</li> </ul> <p>배우관계가 1 ~ 4 가 아닐때, 또는 9 가 아닐때</p> <p>1. 초혼년월이 현년도보다 클때</p> <p>2. 배우관계가 1. 유배우, 2. 사별, 3. 이혼인 경우 초혼년월이 기재되어 있지 않을때</p> <p>1. 성별 2이고 배우관계 미혼이 아닌 가구원이 99, 00 ~ 22 가 아닐때</p> <p>2. 그 이외의 가구원이 공백이 아닐때</p> <p>1. 성별 2이고 배우관계 미혼이 아닌 가구원이 년 000 ~ 985, 월 00 ~ 12, 결과 0 ~ 4 가 아닐때, 년월이 공백이면서 결과도 공백일때 년월이 기재되어 있지 않고 결과도 기재되어 있지 않을때</p> <p>2. 그 이외의 가구원이 공백이 아닐때</p>	

CHECK 항목	정 CODE	COLUMN 위치		ERROR 기 호
		최 초	최 후	
전 입 일 자				
전 입 년	00 ~ 현년	68	69	**
전 입 월	01 ~ 12	70	71	**
전 입 일	01 ~ 31	72	73	**



ERROR CODE 처리	비고
<p>전입년이 00 ~ 현년이 아닐때</p> <p>전입월이 01 ~ 12 가 아닐때</p> <p>전입일이 01 ~ 31 이 아닐때</p>	

CHECK 항목	정 CODE	COLUMN 위치		ERROR 기 호
		최 초	최 후	
<b>2. 변동보고서</b>				
카드번호	2	1	1	*
변동년월	현 년 월	7	9	*
조사구번호	662 개의 조사구	10	13	*
카드구분	1 ~ 7	14	14	*
<b>1) 가구증가</b>				
가구번호	01010 10 1 ~ 249930 30!	15	22	*
가구원수	01 ~ 45	23	24	*
변동일자		25	30	
변 동 년	~ 현년	25	26	*
변 동 월	01 ~ 12	27	28	*
변 동 일	01 ~ 31	29	30	*
구 분	11, 21, 22, 91, 93	40	41	*
전거주지	01111 ~ 53999	31	35	*
이동사유	부호기입요령서 참조	36	39	*

ERROR CODE 처리	비고
<p>카드번호 2가 아닐 때.</p> <p>    금월이 아닐 때.</p> <p>    662개의 조사구가 아닐 때.</p> <p>카드구분이 1~7이 아닐 때.</p> <p>가구번호가 01010101 ~ 24993030이 아닐 때.</p> <p>가구원수가 01~45가 아닐 때.</p> <p>변동년이 현년을 넘었을 때.</p> <p>변동월이 01~12가 아닐 때.</p> <p>변동일이 01~31이 아닐 때.</p> <p>구분이 11, 21, 22, 91, 93이 아닐 때.</p> <p>구분이 22일 때만 해당, 01111~53999가 아닐 때.</p> <p>구분이 22일 때만 해당, 부호기입요령서에 없는 사유</p>	

CHECK 항목	정 CODE	COLUMN 위치		ERROR 기 호
		최 초	최 후	
2) 가구감소				
가구번호	가구증가와 동일	15	22	*
가구원수	"	23	24	*
변동일자				
변동년	~ 현년	25	26	*
변동월	01 ~ 12	27	28	*
변동일자	01 ~ 31	29	30	*
구 분	51, 61, 62, 75, 92, 94	40	41	*
목적지	가구증가 참조	31	35	*
이동사유	"	36	39	*
3) 가구원증가				
가구번호	가구증가와 동일	15	22	*
가구원번호	"	23	24	*
가구주관계	가구표와 동일	45	48	*
성 별	"	49	49	*
생년월일	"	50	56	*

ERROR CODE 처리	비 고
<p>가구증가와 동일</p> <p>  "</p> <p>변동년이 당해년도를 넘었을 때.</p> <p>변동월이 01 ~ 12 가 아닐 때.</p> <p>변동일이 01 ~ 31 이 아닐 때.</p> <p>구분이 51, 61, 62, 75, 92, 94 가 아닐 때.</p> <p>구분이 62, 75 만 해당, 변동보고서 기입항목이 아닐 때.</p> <p>구분이 62, 75 만 해당,               "</p> <p>가구증가와 동일</p> <p>  "</p> <p>가구표와 동일</p> <p>  "</p> <p>  "</p>	

CHECK 항목	정 CODE	COLUMN 위치		ERROR 기 호
		최 초	최 후	
학 력	가구표와 동일	57	59	*
배우관계	//	60	60	*
초혼년월	//	61	64	*
출산계획	//	65	66	*
임신여부	//	67	71	*
구 분	12, 21, 22, 25	87	88	*
전거주지	가구증가 참조	72	76	*
이동사유	//	83	86	*
전입일자		77	82	
전입년	~ 당해년	77	78	*
전입월	01 ~ 12	79	80	*
전입일	01 ~ 31	81	82	*
4) 가구원감소				
가구번호	가구증가 참조	15	22	*
가구원번호	//	23	24	*
변동일자				
변동년	~ 당해년	45	46	*
변동월	01 ~ 12	47	48	*
변동일자	01 ~ 31	49	50	*

ERROR CODE 처리	비 고
<p>가구표와 동일</p> <p>  "</p> <p>  "</p> <p>  "</p> <p>  "</p> <p>구분이 12, 21, 22, 25가 아닐 때.</p> <p>  " 22, 25일 때, 부호기입요령서이외의 코드일 때.</p> <p>  " 22, 25일 때,               "</p> <p>전입년이 당해년을 넘었을 때.</p> <p>전입월이 01~12가 아닐 때.</p> <p>전입일이 01~31이 아닐 때.</p> <p>가구증가와 동일</p> <p>  "</p> <p>변동년이 당해년을 넘었을 때.</p> <p>변동월이 01~12가 아닐 때.</p> <p>변동일이 01~31이 아닐 때.</p>	

CHECK 항목	정 CODE	COLUMN 위치		ERROR 기 호
		최 초	최 후	
구 분	52, 61, 62, 75	60	61	*
목적지	변동부호 기입요령서 참조	51	55	*
이동사유	//	56	59	*
5) 가구표 수정 요구사항				
가구번호	가구증가 참조	15	22	*
가구원 번호	//	23	24	*
가구표란번호	01 ~ 22	25	26	*
수정전 내용		27	33	*
수정후 내용		34	40	*
6) 조사구내이동				
구분	01 ~ 04	15	16	*
이동전 가구 번호	가구증가 참조	17	24	*
이동전 가구 원 번호	01 ~ 49	25	26	*
이동후 가구 번호	가구증가 참조	27	34	*
이동후 가구 원 번호	01 ~ 49	35	36	*
가구주 관계	가구표 참조	37	40	*
거처구분	//	41	41	*
소유관계	//	42	42	*
배우관계	가구표와 동일	43	43	*
초혼년월	//	44	47	*



ERROR CODE 처리	비 고
구분이 52, 61, 62, 75가 아닐 때 구분이 62, 75인 경우 목적지 부호가 아닐 때 구분이 62, 75인 경우 이동사유 부호가 아닐 때	
가구증가와 동일	
가구표란 번호가 01 ~ 22가 아닐 때 가구표란 번호가 11, 17일때 수정후 내용이 공백일때 11,17 이외의 가구표란 번호일 때 수정후 내용이나 수정전 내용이 공백인 경우 구분이 01 ~ 04가 아닐 때	
가구증가와 동일 " , 구분이 01이고 공백이 아닐 때	
가구증가와 동일 " , 구분이 01이고 공백이 아닐 때 가구표와 동일, 구분이 02, 03, 04만 해당	
" , 구분이 01일 때 해당	
" , "	
" , 구분이 01이 아닐 때	
" "	

CHECK 항목	정 CODE	COLUMN 위치		ERROR 기 호
		최 초	최 후	
7) 임신기록및결과				
가구번호	가구증가와 동일	15	22	*
가구원번호	"	23	24	*
출산예정				
출산 년	80 ~ 당해 년	45	46	*
출산 월	01 ~ 12	47	48	*
결 과	0 ~ 9	49	49	*

ERROR CODE 처리	비 고
<p>가구증가와 동일</p> <p>  "</p> <p>출산년이 80 ~ 당해년이 아닐때, 결과가 공백일때</p> <p>출산월이 01 ~ 12가 아닐 때,       "</p> <p>출산예정일이 공백일 때</p>	

CHECK 항목	정 CODE	COLUMN 위치		ERROR 기 호
		최 초	최 후	
<b>3. 표본조사표</b>				
카드번호	3	1	1	*
일련번호	가구표 참조	2	6	*
조사구번호	//	7	10	*
가구번호	//	11	18	*
카드구분	1 ~ 4	19	19	*
1) 출 생				
출생 년	85 ~ 86, 99	62	63	*
출생 월	01 ~ 12, 99	64	65	*
출생 일	01 ~ 31, 99	66	67	*
성 별	1 ~ 2, 9	68	68	*
출생당시 모의연령	15 ~ 55, 99	69	70	*
출산순위	01 ~ 10, 99	71	72	*
임신기간	28 ~ 43, 99	73	74	*
모의교육정도	000 ~ 183, 999	75	77	*
조 산 자	1 ~ 3, 9	78	78	*

ERROR CODE 처리	비 고
<p>카드번호가 3이 아니면 ERROR</p> <p>가구포 참조</p> <p>//</p> <p>//</p> <p>카드구분이 1~4가 아닐 때</p> <p>출생년도가 85~86, 99가 아닐 때</p> <p>출생월이 01~12, 99가 아닐 때</p> <p>출생일이 01~31, 99가 아닐 때</p> <p>성별이 1~2, 9가 아닐 때</p> <p>출생당시 모의연령이 15~55, 99가 아닐 때</p> <p>출산순위가 01~10, 99가 아닐 때</p> <p>임신기간이 28~43, 99가 아닐 때</p> <p>모의교육정도가 000~183, 999가 아닐 때</p> <p>조산자가 1~3, 9가 아닐 때</p>	

CHECK 항목	정 CODE	COLUMN 위치		ERROR 기 호
		최 초	최 후	
2) 사 망				
가구원번호	01 ~ 49	40	41	*
성 별	1 ~ 2, 9	42	42	*
생년월일	870 ~ 986, 999(년)	43	45	*
	01 ~ 12, 99(월)	46	47	*
	01 ~ 31, 99(일)	48	49	*
사망년월일	985 ~ 986, 999(년)	50	52	*
	01 ~ 12, 99(월)	53	54	*
	01 ~ 31, 99(일)	55	56	*
사망자의교육정도	000 ~ 183, 999	57	59	*
직 업	00 ~ 07, 99	60	61	*
혼인상태	1 ~ 4, 9	62	62	*
진단자	1 ~ 3, 9	63	63	*
사망원인	11, 21, 29, 31, 41, 49, 51,	64	65	*
	61, 71, 81, 99			

ERROR CODE 처리	비고
가구원번호가 01 ~ 49 가 아닐때	
성별이 1 ~ 2 , 9 가 아닐때	
생년이 870 ~ 986, 999 가 아닐때	
생월이 01 ~ 12, 99 가 아닐때	
생일이 01 ~ 31, 99 가 아닐때	
사망년이 985 ~ 986, 999 가 아닐때	
사망월이 01 ~ 12, 99 가 아닐때	
사망일이 01 ~ 31, 99 가 아닐때	
사망자의 교육정도가 000 ~ 183, 999 가 아닐때	
직업이 00 ~ 07, 99 가 아닐때	
혼인상태가 1 ~ 4 , 9 가 아닐때	
진단자가 1 ~ 3 , 9 가 아닐때	
사망원인이 11, 21, 29, 31, 41, 49, 51, 61, 71, 81,	
99 가 아닐때	

CHECK 항목	정 CODE	COLUMN 위치		ERROR 기 호
		최 초	최 후	
3) 혼 인				
부의 연령	15 ~ 50	42	43	*
부의 교육정도	000 ~ 183, 999	44	46	*
처의 연령	15 ~ 50	69	70	*
처의 교육정도	000 ~ 183, 999	71	73	*
혼인년월	84 ~ 86, 99(년)	74	75	*
	01 ~ 12, 99(월)	76	77	*
4) 이 혼				
부의 연령	15 ~ 98	40	41	*
처의 연령	15 ~ 98	62	63	*
생존자녀수	0 ~ 10	64	65	*
혼인년	20 ~ 85, 99	66	67	*
혼인월	01 ~ 12, 99	68	69	*
발생년	86	70	71	*
발생월	01 ~ 12, 99	72	73	*



ERROR CODE 처리	비고
부의 연령이 15 ~ 50 이 아닐때	
부의 교육정도가 000 ~ 183 이 아닐때	
처의 연령이 15 ~ 50 이 아닐때	
처의 교육정도가 000 ~ 183 이 아닐때	
혼인년이 84 ~ 86, 99 가 아닐때	
혼인월이 01 ~ 12, 99 가 아닐때	
부의 연령이 15 ~ 98 이 아닐때	
처의 연령이 15 ~ 98 이 아닐때	
생존자녀수가 0 ~ 10 이 아닐때	
혼인년이 20 ~ 85, 99 가 아닐때	
혼인월이 01 ~ 12, 99 가 아닐때	
발생년이 86 이 아닐때	
발생월이 01 ~ 12, 99 가 아닐때	

## 4. CRT 조작 방법

**1. 작업준비 ( CRT )**

- (1) 전원 PLUG 연결
- (2) NORMAL / TEST SWITCH를 NORMAL에 위치
- (3) POWER SWITCH ON
  - 1 부분을 누른다.
  - LIGHT 3에 불이 들어온다.
  - LIGHT1에 불이 들어온다.
  - 잠시후 LIGHT 2에 불이 들어온다.
- (4) VM/ 3278 ONLINE 화면이 나타난다.

```

                @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
                @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
                @E                                     @E
                @E      ===== @E
                @EE     ===== @EE
                @EE     === @EE
                @EE     ===== @EE
                @EE     === @EE
                @EE     ===== @EE
                @EE     ===== @EE
                @EE     ===== @EE
                @E E.F.B COMPUTER CENTER ==> ACF/VTAM V.2.1 @E
                @E
                @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
                @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
                |X| MVS EXPRESS +8502 |X|
                +-----+
                | *** PLEASE ENTER TSO,CICS,CICS3 | /E|
                | OR YOUR LOGON COMMAND *** | /EE/
                | =====> | /EE/
                +-----+ /EE/
                @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
            
```

윗화면 =====> TSO를 KEY IN하고 ENTER KEY를 누르면  
 다음과 같은 화면이 나온다.

```

IKJ56700A ENTER USERID -
    
```

USER의 고유번호 A 33을 KEY IN하고 ENTER KEY를 누른다.

```
----- VS2 REL 3.8 TIME SHARING OPTION -----
PF1/PF13 ==> HELP   PF3/PF15 ==> LOGOFF   PA1 ==> ATTENTION   PA2 ==> RESHOW
YOU MAY REQUEST SPECIFIC HELP INFORMATION BY ENTERING A '?' IN ANY ENTRY FIELD.
ENTER LOGON PARAMETERS BELOW:                RACF LOGON PARAMETERS:
```

```
USERID      ==> A33
PASSWORD    ==>
NEW PASSWORD ==>
PROCEDURE   ==> BOSUSER
GROUP IDENT ==>
ACCT NMBR   ==> 29
SIZE        ==> 5120
PERFORM     ==>
COMMAND     ==>
```

ENTER AN 'S' BEFORE EACH OPTION DESIRED BELOW:

```
-NOMAIL      -NONOTICE      -RECONNECT      -DIDCARD
```

PASSWORD 위치에 USER의 비밀번호를 KEY IN하고 ENTER KEY를 누른다.

```
ICH70001I A04      LAST ACCESS AT 14:21:15 ON THURSDAY, APRIL 17, 1986
IKJ56455I A04 LOGON IN PROGRESS AT 13:58:40 ON APRIL 18, 1986
IKJ56951I NO BROADCAST MESSAGES
```

잠시후 별표가 나온다.

\*\*\*

ENTER KEY를 누르면 다음과 같은 화면이 나온다.

```
----- ISPF/PDF PRIMARY OPTION MENU -----
OPTION ==>
0 ISPF PARMS - SPECIFY TERMINAL AND USER PARAMETERS  USERID - A04
1 BROWSE     - DISPLAY SOURCE DATA OR OUTPUT LISTINGS TIME - 13:58
2 EDIT      - CREATE OR CHANGE SOURCE DATA          TERMINAL - 3278
3 UTILITIES - PERFORM UTILITY FUNCTIONS              PF KEYS - 24
6 COMMAND   - ENTER TSO COMMAND OR CLIST
8 SDSF UTILITY- SPOOL DATA SEARCH FACILITY FUNCTIONS
T TUTORIAL  - DISPLAY INFORMATION ABOUT ISPF/PDF
X EXIT     - TERMINATE ISPF USING LOG AND LIST DEFAULTS
```

ENTER END COMMAND TO TERMINATE ISPF.

화면에 있는 2번숫자를 OPTION ===> 위치에 KEY IN하고  
 ENTER KEY를 누르면 다음과 같은 화면이 나온다.

```

----- EDIT - ENTRY PANEL -----
COMMAND ===>

ISPF LIBRARY:
  PROJECT ===> A33
  GROUP   ===> PROGRAM   ===>           ===>           ===>
  TYPE    ===> LIBRARY
  MEMBER  ===>           (BLANK FOR MEMBER SELECTION LIST)

OTHER PARTITIONED OR SEQUENTIAL DATA SET:
  DATA SET NAME ===>
  VOLUME SERIAL  ===>           (IF NOT CATALOGED)

DATA SET PASSWORD ===>           (IF PASSWORD PROTECTED)

PROFILE NAME     ===>           (BLANK DEFAULTS TO DATA SET TYPE)

INITIAL MACRO    ===>

MIXED MODE      ===> NO           (SPECIFY YES OR NO)

FORMAT NAME     ===>

MEMBER =====>
  
```

① 방법

ENTER KEY만 누르면 FILE NAME만 있는 화면이 나온다.

```

EDIT --- A33.PROGRAM.LIBRARY -----
COMMAND ===>
  
```

NAME	VV.MM	CREATED	LAST MODIFIED	SIZE	INIT	MOD	CSR
SVHWAA52	01.00	86/01/09	86/01/09 15:25	32	32	0	A33
SVHWBB33	01.02	85/01/17	86/02/15 10:03	41	47	0	A33
SVHWCA52							
SVHWCB33	01.01	85/01/30	86/01/09 15:25	78	78	0	A33
SVHWCC52	01.03	86/01/09	86/02/24 18:04	32	223	1	A33
SVHWCI52	01.00	86/01/09	86/01/09 15:26	56	56	0	A33
SVHWCF33	01.02	86/01/09	86/02/24 18:03	37	43	10	A33
SVHWDD52	01.00	86/01/09	86/01/09 15:26	59	59	0	A33
SVHWDI52	01.25	85/01/14	86/01/09 15:27	77	65	0	A33
SVHWDI93	01.09	85/03/15	86/02/24 17:39	68	80	0	A33
SVHWDV93	01.07	85/01/24	86/02/20 16:45	56	56	0	A33
SVHWEA93	01.29	85/01/15	86/01/09 15:27	265	265	0	A33
SVHWEF93	01.17	85/01/15	86/01/09 15:28	268	268	0	A33
SVHWGB33	01.00	86/01/09	86/01/09 15:28	208	208	0	A33
SVHWGM33	01.00	86/01/09	86/01/09 15:28	217	217	0	A33
SVHWA52	01.15	85/01/18	86/01/09 15:28	172	178	0	A33
SVHWMB52	01.07	85/01/18	86/01/09 15:28	219	225	0	A33
SVHWMM33	01.00	86/01/09	86/01/09 15:29	172	172	0	A33
SVHWM152	01.14	85/01/22	86/01/09 15:29	126	132	0	A33
SVHWM252	01.17	85/01/22	86/01/09 15:29	524	489	0	A33
SVHWPA33	01.00	86/01/09	86/01/09 15:29	95	95	0	A33

윗화면에서는 필요한 FILE NAME 앞에 S를 KEY IN하고 ENTER KEY를 누르면 2. 방법과 같은 화면이 나온다.

② 방 법

직접 JOB NAME ( A 33 VHWT 8 )을 KEY IN하고 ENTER KEY를 누르면 USER가 원하는 PROGRAM이 나온다.

```

EDIT ---- A33.PROGRAM.LIBRARY(BYUNTAB) -- 01.05 ----- COLUMNS 001 072
COMMAND ==(>
***** ***** TOP OF DATA *****
==MSG> -CAUTION- PROFILE CHANGED TO "NUMBER ON STD" (FROM "NUMBER OFF").
==MSG> DATA HAS VALID STANDARD NUMBERS.
000001 //A33VHWT8 JOB CLASS=A,MSGCLASS=X,NOTIFY=A33
000002 //BYUNTAB EXEC PLIFCLG
000003 //PLI.SYSIN DD *
000004 * PROCESS S,GS,NEST,OPT(2);
000005 LEE: PROC OPTIONS(MAIN);
000006 DCL TPI FILE RECORD INPUT;
000007 DCL 1 TA,
000008 2 CD CHAR(1),
000009 2 CR CHAR(5),
000010 2 YR CHAR(3),
000011 2 GS CHAR(4),
000012 2 ER CHAR(1),
000013 2 SS CHAR(8),
000014 2 EE PIC'(2)9',
000015 2 CARD CHAR(76);
000016 DCL TAB(3) FLOAT DEC(9) INIT(0);
000017 DCL TOT(3) FLOAT DEC(9) INIT(0);
000018 DCL CNT CHAR(4) INIT('0');
000019 DCL SW CHAR(1) INIT('0');

```

윗 화면상태에서 PROGRAM을 USER가 수정 및 보완을 할 수 있으며 작업을 하기 위하여 SUB도 할 수 있다.

모든 작업이 끝나면 =X를 KEY IN하고 ENTER KEY를 누르면 다음과 같은 화면이 나온다.

```

----- SPECIFY DISPOSITION OF LOG DATA SET -----
COMMAND ===>

LOG OPTIONS FOR THIS SESSION          NO LIST DATA SET THIS SESSION
-----
PROCESS OPTION    ===>
SYSOUT CLASS     ===> A
LOCAL PRINTER ID ===>

VALID PROCESS OPTIONS:
  PD - PRINT DATA SET AND DELETE      K - KEEP DATA SET (WITHOUT PRINTING)
                                         D - DELETE DATA SET (WITHOUT PRINTING)

INSTRUCTIONS:
  PRESS ENTER KEY TO COMPLETE ISPF TERMINATION.
  ENTER END COMMAND TO RETURN TO THE PRIMARY OPTION MENU.

JOB STATEMENT INFORMATION: (REQUIRED FOR SYSTEM PRINTER)
===> //A33A      JOB (ACCOUNT), 'NAME'
===> /**
===> /**
===> /**

```

화면에서 PROCESS OPTION===> 위치에 D를 KEY IN 하고  
ENTER KEY 를 누른다.

```

A33.SPFLOG1.LIST HAS BEEN DELETED.
READY

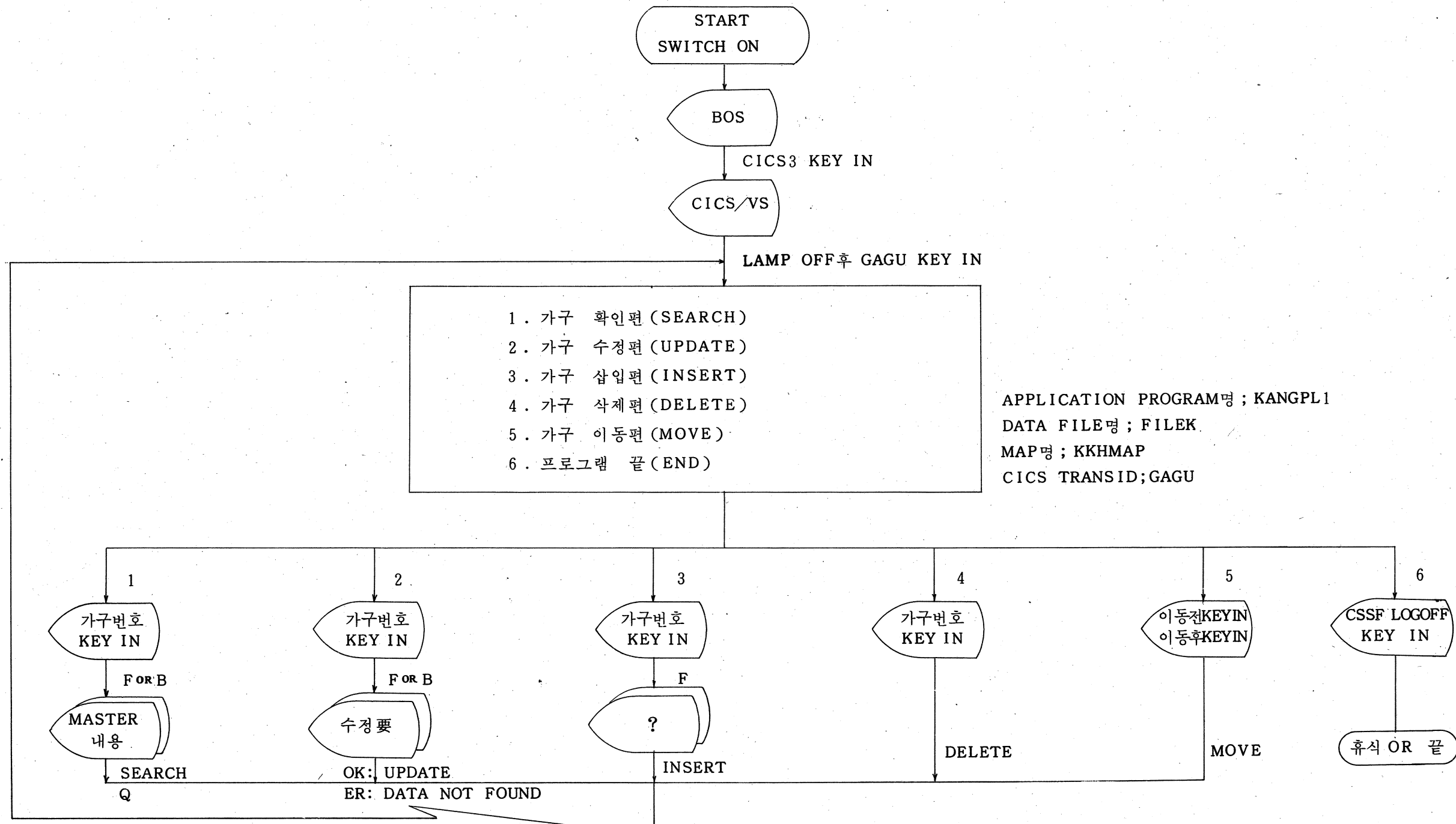
```

LOGOFF를 KEY IN 하고 ENTER KEY 를 누르면 처음과 같은 화  
면이 나오면서 모든 작업이 끝난다.

5. CICS 사용 방법  
(CUSTOMER INFORMATION  
CONTROL SYSTEM) 이용방법



# 1. 가구관리 MASTER ON-LINE SYSTEM FLOW



＊ **CICS3** 조작초 주의사항

TRANSACTION ID ( GAGU ) 를 KEY IN 하기전에 GAGU 칠자리에  
CEMT I DATA 를 KEY IN 하여 사용하고자 하는 FILE이 OPEN  
되어있나 확인하여야 한다.

GAGU의 FILE인 FILEK 를 확인한다.

OPE ENA 상태일 경우에는 OPEN된 상태이며,

CLO DIS 상태일 경우는 CLOSE된 상태이다.

OPEN 시키고자 할 때는 OPE ENA 를 CLOSE 시키고자 할 때는  
CLO DIS 를 만든다.

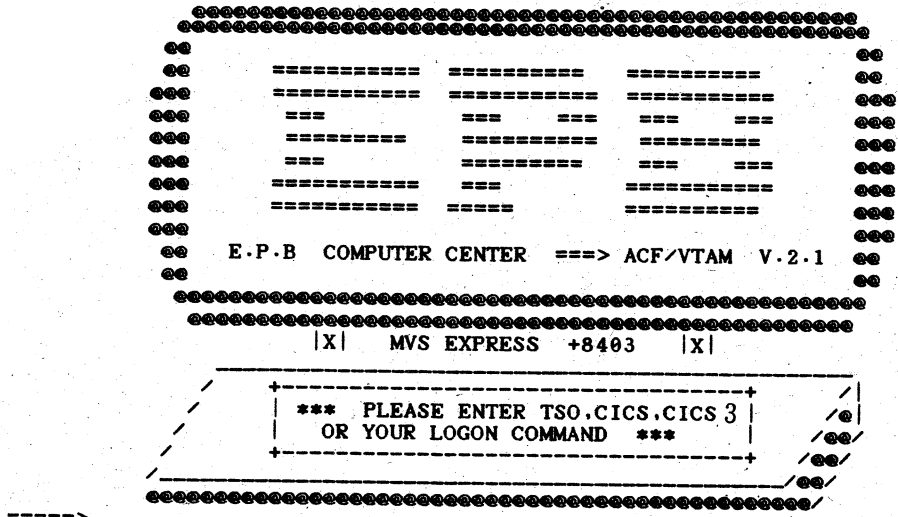
일단 확인한 후 PF3 KEY 를 누르고 CLEAR KEY 를 누르면 처음  
GAGU 를 칠 수 있는 상태가 된다.

＊ **BATCH** 작업을 수행할 경우

FILEK가 CLOSE 상태가 되어야 하므로 CICS3 를 조작하여 CLO  
DIS 상태로 만든후 BATCH 작업을 수행한다.

**1. CICS3** 조작방법

- (1) 전원 PLUG 연결
- (2) 뒷면의 POWER SWITCH ON
- (3) 잠시후 다음과 같은 화면이 나타난다.



윗화면 ==> CICS3 를 KEY IN 하고 ENTER KEY 를 누르면 다음과 같은 화면이 나온다.

\*\*\*DFH2312 WELCOME TO CICS/VS \*\*\* 16:49:26

```

CCCCC   IIIII   CCCCC   SSSSSS   VVVV VVVV   SSSSSS
CCCCCCC IIIII CCCCCC  SSSSSSSS  VVV VVV  SSSSSSSS
CCC CC   III   CCC CC  SSSS SS   VVV VVV  SSSS SS
CCC      III   CCC      SSSS      ***   VVV VVV  SSSS
CCC CC   III   CCC CC  SS SSSS    ***   VVVVVV  SSSS
CCCCCCC IIIII CCCCCC  SSSSSSSS  VVVVV  SS SSSS
CCCCC   IIIII CCCCC   SSSSSS     VVV   SSSSSSSS
                                     VV   SSSSSS

```

- 윗화면이 나온후 좌측상단의 LAMP에 불이 꺼진후 HOME(CLR) KEY 를 누르면 화면이 지워지고 CURSUR가 깜박깜박하면 GAGU 라는 TRANSACTION ID를 KEY IN하고 ENTER KEY를 치면

잠시후 가구관리전산화 화면이 나온다.

## 가 구 관 리 전 산 화

아래번호중 택일하여 KEY IN

- 1 - - - > 가구 확인편 ( S E A R C H )
- 2 - - - > 가구 수정편 ( U P D A T E )
- 3 - - - > 가구 삽입편 ( I N S E R T )
- 4 - - - > 가구 삭제편 ( D E L E T E )
- 5 - - - > 가구 이동편 ( M O V E )
- 6 - - - > 프로그램 끝 ( E N D )

---->

윗화면에서 USER가 원하는 숫자1를 KEY IN하고 ENTER KEY  
를 누르면 다음과 같은 화면이 나온다.

## 가 구 확 인 편

----> ?.....?

위 화면에서 KEY ( 조사구번호와 가구번호 ) 12 자리를 KEY IN 하고 ENTER KEY를 누르면 다음과 같은 화면이 나온다.

```

-----가 구 사 항-----
입련   1   2-4   6 7 8 9   22   5
번호   조사   가   구   거소구가   부적   행   정   주   소
번호   번호   번   호   처유역구   격   구   역
00104  0001  04030101  2111           111735  신일연립  다동  1 0 5 호  1
-----가 구 원 사 항-----
10   11   12   13   14   15   16   17   18   19   20   21
번호   성   명   관계   성   생   년   배   초   출   임   전   비   교
번호   명   관   성   월   일   학   우   초   산   신   입   자   고   체
01   오   태   기   0100  1   9560402  122  1   8311           840801
02   최   호   순   0200  2   9590727  122  1   8311   11  85011  840801
03   오   재   식   0500  1   9340228  073  1   5504           840801
04   조   황   자   0500  2   9360611  062  1   5504   00           840801
05   오   만   기   0400  1   9581028  122  4           840801
06   오   덕   기   0400  1   9610824  122  4           840801
07   오   동   기   0400  1   9640511  122  4           840801

```

===>

위 화면상태에서 ===> 위치에서

F : ( FORWARD )

B : ( BACKWARD )

Q : ( QUIT )

할 수 있다.

원하는 가구확인이 끝났으면 Q를 KEY IN하고 ENTER KEY를 누르면 가구확인편 화면이 나온다.

가 구 확 인 편

----> ?.....?

이상태에서 더 이상 다른 가구의 확인이 필요없을 때 HOME (CLR) KEY를 누르면 다음과 같은 화면이 나온다.

MAP NOT FOUND ..... ENTER TO CONTINUE

ENTER KEY를 누르면 가구관리 전산화 화면이 나온다.

가 구 관 리 전 산 화

아래번호중 택일하여 KEY IN

- 1 - - - > 가구 확인편 ( S E A R C H )
- 2 - - - > 가구 수정편 ( U P D A T E )
- 3 - - - > 가구 삽입편 ( I N S E R T )
- 4 - - - > 가구 삭제편 ( D E L E T E )
- 5 - - - > 가구 이동편 ( M O V E )
- 6 - - - > 프로그램 끝 ( E N D )

----> 2

위화면에서 USER가 원하는 숫자 2를 KEY IN하고 ENTER KEY  
를 누르면 다음과 같은 화면이 나온다.

가 구 수 정 편

----> ?.....?

조사구번호와 가구번호 (KEY 12자리)를 KEY IN하고 ENTER  
KEY를 누른다.

-----가 구 사 항-----

입력 번호	1 조사 번호	2-4 가 구 번호	6 7 8 9 거소구가 처유역구	22 부적 격 구	행 정 역	5 주 소
00104	0001	04030101	2111		111735	신일연립 다동 105호 1

-----가 구 원 사 항-----

10 번 호	11 성 명	12 관 계	13 성	14 생 월	15 년 일	16 학 력	17 배 우	18 초 혼	19 출 생 일	20 신 입 여 부	21 전 입 일 자	22 비 고	23 체 고
01	오태기	0100	1	9560402	122	1	8311				840801		
02	최효순	0200	2	9590727	122	1	8311	11	85011		840801		
03	오재식	0500	1	9340228	073	1	5504				840801		
04	조황자	0500	2	9360611	062	1	5504	00			840801		
05	오만기	0400	1	9581028	122	4					840801		
06	오덕기	0400	1	9610824	122	4					840801		
07	오동기	0400	1	9640511	122	4					840801		

====> F

위 화면 상태에서 DATA 수정을 할 수 있다. ( 단, === > 위치에 F 또는 B가 있을시에만 수정이 가능하다 )

DATA 수정이 끝난후 ENTER KEY를 누르면 가구원이 7人以上인 경우에 다음가구원 화면이 나오고 7人以下인 경우엔 가구수정편 화면이 나온다.

### 가 구 수 정 편

----> ?.....?

이 화면에서 다음 수정사항이 있으면 KEY를 치고 수정이 끝난후 즉 update 할 DATA가 없으면 가구수정편 화면에서 HOME (CLR) KEY를 누르면 다음과 같은 MESSAGE가 나온다.

MAP NOT FOUND ..... ENTER TO CONTINUE

후에 ENTER KEY를 치면 화면은 다시 가구관리 전산화 화면으로 가게 되는 것이다.



## 가 구 관 리 전 산 화

아래번호중 택일하여 KEY IN

- 1 - - - > 가구 확인편 ( S E A R C H )
- 2 - - - > 가구 수정편 ( U P D A T E )
- 3 - - - > 가구 삽입편 ( I N S E R T )
- 4 - - - > 가구 삭제편 ( D E L E T E )
- 5 - - - > 가구 이동편 ( M O V E )
- 6 - - - > 프로그램 끝 ( E N D )

----> 3

위화면에서 USER가 원하는 숫자 3을 KEY IN하고 ENTER KEY를 누르면 다음과 같은 화면이 나온다.

## 가 구 삽 입 편

----> ?.....?

위화면에서 USER가 원하는 조사구번호와 가구번호(KEY)를 치면 다음과 같은 화면이 나온다. 이때의 KEY는 기존에 없는 KEY이어야 할 것이다.

```

-----가 구 사 항-----
입력   1   2-4   6 7 8 9   22   5
번호   조사  가 구   거 소 구 가   부 적   행   정   주   소
번호   번호  번 호   처 유 역 구   격   구   역
00104  0001  04030101  2111           111735  신일연립 다동 105호 1
-----가 구 원 사 항-----
10   11   12   13   14   15   16 17   18 19   20   21
번호  성 명   관 계   성   생   년   배   출   입   신   전   입   비   교   체
호    성 명   관 계   성   월   일   학 력   우   초   혼   산   여   부   일   자   고   체

```

위화면에서 각각의 사항(가구, 또는 가구원 사항)을 INSERT한다. INSERT할 RECORD가 7명을 넘었을때 ==> 에 F를 KEY IN 하면 가구사항의 INSERT된 내용과 BLANK 가구원 사항 화면이 나타나게 될 것이다. 여기에서 다음(8번째 가구원부터) INSERT시킨다. 끝난후 ENTER KEY를 누르면 다음과 같은 MESSAGE가 나온다.

INSERT COMPLETED ..... PRESS ENTER TO CONTINUE

여기서 ENTER KEY를 누르면 다시 가구관리 전산화 화면이 나온다.

가 구 관 리 전 산 화

- 아래번호중 택일하여 KEY IN
- 1 - - - > 가구 확인편 (SEARCH)
  - 2 - - - > 가구 수정편 (UPDATE)
  - 3 - - - > 가구 삽입편 (INSERT)
  - 4 - - - > 가구 삭제편 (DELETE)
  - 5 - - - > 가구 이동편 (MOVE)
  - 6 - - - > 프로그램 끝 (END)

여기에서 USER가 원하는 KEY 4번을 누르면 다음과 같은 화면이 나온다.

가 구 삭 제 편

가 구 삭 제 시 엔 가 구 원 번 호 0 0 K E Y I N

---> ?.....?

위 화면에서 삭제할 KEY는 조사구번호와 가구번호외에 가구원번호를 포함한 KEY이다. 가구원번호가 00일 경우에는 가구전체가 한꺼번에 삭제되므로 유의하여야만 한다. 다음은 KEY를 친후 나타난 화면이다.

DELETE COMPLETED ..... PRESS ENTER TO CONTINUE

이 MESSAGE는 KEY중에서 끝 두자리(즉 가구원 번호)가 00일 경우엔 가구전체의 DATA가 삭제되었음을 의미하는 것이고 00이 아닌 경우엔 모가구의 한가구원이 삭제되었음을 의미한다.

ENTER KEY를 치면 다시 가구관리 전산화 화면이 나온다.

가 구 관 리 전 산 화

아래번호중 택일하여 KEY IN

- 1 - - - > 가구 확인편 ( S E A R C H )
- 2 - - - > 가구 수정편 ( U P D A T E )
- 3 - - - > 가구 삽입편 ( I N S E R T )
- 4 - - - > 가구 삭제편 ( D E L E T E )
- 5 - - - > 가구 이동편 ( M O V E )
- 6 - - - > 프로그램 끝 ( E N D )

----> 5

위화면에서 USER가 원하는 5번 KEY를 KEY IN 한 후에 ENTER KEY를 누르면 다음과 같은 가구 이동편 화면이 나온다.

가 구 이 동 편

----> ?.....?

----> ?.....?

처음의 ----> ? .....? 은 이동전 가구번호이고 다음은 이동후 가구번호이다. 이 두자리에서 전자는 이미 존재하는 KEY이고 후자는 새로운 KEY이다. 전자의 KEY를 가진 가구의 내용은 후자의 KEY로 INSERT된다. 끝난 후에 ENTER KEY를 누르면 아

래와 같은 MESSAGE

MOVE COMPLETED ..... PRESS

ENTER TO CONTINUE 가 나온다.

이 의미는 이동전 가구의 모든 DATA가 삭제되는 것이 아니라 같은 DATA가 KEY만 다를 뿐 COPY되어져 양가구의 DATA는 같은 것들이지만 유고사항의 ALTERNATE KEY가 다르게 들어가고 있다.

이때 ENTER KEY를 치면 다시 가구관리 전산화 화면이 나온다.

### 가 구 관 리 전 산 화

아래번호중 택일하여 KEY IN

- 1 - - - > 가구 확인편 ( S E A R C H )
- 2 - - - > 가구 수정편 ( U P D A T E )
- 3 - - - > 가구 삽입편 ( I N S E R T )
- 4 - - - > 가구 삭제편 ( D E L E T E )
- 5 - - - > 가구 이동편 ( M O V E )
- 6 - - - > 프로그램 끝 ( E N D )

--> 6

모든 작업이 끝난후 6번을 KEY IN하면 다음과 같은 MESSAGE  
가 나온다.

CICS ENDED ..... CSSF LOGOFF KEYIN!!!

CiCS3를 OFF하기 위하여

CSSF LOGOFF를 KEY IN하고 ENTER KEY를 누르면 맨 처음 화면이 나오며 CiCS3는 LOG OFF되어 모든 작업은 끝난다.

## 6. SOURCE PROGRAM LIST

A33,86.055,17:16:00,A33.PROGRAM.LIBRARY(GASORT)

```
//A33VHUSA JOB CLASS=A,MSGCLASS=X,NOTIFY=A33,TYPRUN=HOLD 00000100
//STEP1 EXEC PGM=SORT,PARM='SIZE(MAX)' 00000200
//SYSOUT DD SYSOUT=* 00000300
//SYSPRINT DD SYSOUT=* 00000400
//SORTIN DD DSN=S93VHTTT,UNIT=TAPE,DISP=(OLD,KEEP),LABEL=(,NL), 00000500
// DCB=(RECFM=F,LRECL=100,BLKSIZE=100), 00000600
// VOL=SER=TN6111 00000700
//SORTOUT DD DSN=A33.VH.TEM(SVHS1093),UNIT=DISK,DISP=(OLD,KEEP), 00000800
// DCB=(RECFM=FB,LRECL=100,BLKSIZE=4000), 00000900
// VOL=SER=BOS005 00001000
//SORTWK01 DD UNIT=DISK,SPACE=(CYL,150),VOL=SER=SORTWK 00001100
//SYSIN DD * 00001200
SORT FIELDS=(7,12,A,19,2,A),FORMAT=BI,FILSZ=E120000 00001300
/* 00001400
// 00001500
```



A33,86.055,17:18:26,A33.PROGRAM.LIBRARY(GAEDIT)

```
//A33VHWEA JOB CLASS=A,NOTIFY=A33,MSGCLASS=X 00000100
//GACHECK EXEC PLIFCLG,REGION=2048K 00000200
//PLI.SYSIN DD * 00000300
* PROCESS S,GS; 00000400
RIVER: PROC OPTIONS(MAIN); 00000500
  DCL TPI FILE RECORD INPUT ; 00000600
  DCL TTI(40) CHAR(100); 00000700
  DCL 1 TI, 00000800
    2 CARD CHAR(1), 00000900
    2 ILNO CHAR(5), 00001000
    2 JOSA PIC '(4)9', 00001100
    2 GANO CHAR(10), 00001200
    2 GU CHAR(1), 00001300
    2 SOU CHAR(1), 00001400
    2 GUK CHAR(1), 00001500
    2 GAG CHAR(1), 00001600
    2 SAYU CHAR(2), 00001700
    2 ADDR CHAR(40), 00001800
    2 FIL1 CHAR(34); 00001900
  DCL 1 TC, 00002000
    2 CARD1 CHAR(18), 00002100
    2 INO CHAR(2), 00002200
    2 NAME CHAR(20), 00002300
    2 GITA1, 00002400
      3 GAJU1 CHAR(2), 00002500
      3 GAJU2 CHAR(2), 00002600
      3 SEX CHAR(1), 00002700
      3 BIRY PIC '(3)9', 00002800
      3 BIRM PIC '(2)9', 00002900
      3 BIRD PIC '(2)9', 00003000
      3 HAK1 CHAR(2), 00003100
      3 HAK2 CHAR(1), 00003200
      3 BAWO CHAR(1), 00003300
      3 CHY PIC '(2)9', 00003400
      3 CHM PIC '(2)9', 00003500
      3 CHN CHAR(2), 00003600
      3 IMSY CHAR(2), 00003700
      3 IMSM CHAR(2), 00003800
      3 IMSR CHAR(1), 00003900
      3 JUNY CHAR(2), 00004000
      3 JUNM CHAR(2), 00004100
      3 JUND CHAR(2), 00004200
      3 BLK CHAR(27); 00004300
  DCL TII DEF TI CHAR(100); 00004400
  DCL TCC DEF TC CHAR(100); 00004500
  DCL TIT CHAR(100) INIT('...5...10...15...20... 00004600
25...30...35...40...45...50...55...60...65...70...75...80...85...90...900004700
5...100'); 00004800
  DCL SKEY CHAR(12) INIT((12)'0'); 00004900
  DCL ASK CHAR(20) INIT((20)'*'); 00005000
  DCL (SW,ESW) CHAR(1) INIT('0'); 00005100
  DCL EOF BIT(1), 00005200
  YES BIT(1) INIT('1'B); 00005300
  NO BIT(1) INIT('0'B); 00005400
  DCL ERR CHAR(100) INIT(''); 00005500
  DCL I PIC '999' INIT(0); 00005600
  DCL DJOSA(572) PIC '(4)9' INIT 00005700
(0001,0002,0003,0005,0006,0007,0009,0010,1011,2011,0013,0014, 00005800
0015,0017,0018,0019,0021,0022,0023,0025,0026,0027,0029,0030, 00005900
0031,0033,0034,0035,0037,0038,0039,0041,0042,0043,0045,0046, 00006000
0047,0049,0050,0051,0053,0054,0055,0057,0058,0059,0061,0062, 00006100
0063,0065, 00006200
0066,0067,0069,0070,0071,0073,0074,0075,0077,0078, 00006300
```

```

0079,0081,0082,0083,0085,0086,0087,0089,0090,0091,      00006400
0093,0094,0095,0097,0098,0099,0101,0102,0103,0105,      00006500
0106,0107,0109,0110,0111,0113,0114,0115,0117,0118,      00006600
0119,0121,0122,0123,0125,0126,0127,0129,0130,0131,      00006700
0133,0134,0135,0137,0139,0141,0142,0143,0145,0147,      00006800
0149,0150,0151,0153,0154,0155,0157,0158,0159,0161,      00006900
0162,0163,0165,0166,0167,0169,0170,0171,0173,          00007000
0174,0175,0177,0178,0179,0181,0182,0183,0185,0186,      00007100
0187,0189,0190,0191,0193,0194,0195,0197,0198,0199,      00007200
0201,0202,0203,0205,0206,0209,0210,0211,0213,0214,      00007300
0215,0217,0218,0219,0221,0222,0223,0225,0226,0227,      00007400
0229,0230,0250,0251,0253,0254,0255,0257,0258,0259,      00007500
0261,0262,0263,0265,0266,0267,0269,0270,0271,0273,      00007600
0274,0275,0277,0278,0279,0281,0282,0283,0285,0286,      00007700
0287,0289,0290,0291,0293,0294,0295,0297,0298,0299,      00007800
0301,0302,0303,0305,0306,0307,0309,0310,0311,0313,      00007900
0314,0315,0317,0318,0319,0321,0322,0323,0325,0326,      00008000
0327,0329,0330,0331,0333,0334,0335,0337,0351,0353,      00008100
0354,0355,0357,0358,0359,0361,0362,0363,0365,0366,      00008200
0367,0369,0370,0371,0373,0374,0375,0377,0378,0379,      00008300
0381,0382,0383,0385,0386,0387,0389,0390,0391,0393,      00008400
0394,0395,0397,0398,0399,0401,0403,0405,0406,0407,      00008500
0409,0410,0411,0413,0414,0415,0417,0418,0419,0421,      00008600
0422,0423,0425,0426,0427,0429,0430,0431,0433,0434,      00008700
0502,0503,0505,0506,0507,0509,0510,0511,0513,0514,      00008800
0515,0517,0518,0519,0521,0522,0523,0525,0526,0527,      00008900
0529,0530,0531,0533,0534,0535,0537,0538,0539,0541,      00009000
0542,0543,0545,0546,0549,0550,0551,0553,0554,0555,      00009100
0557,0558,0559,0561,0562,0563,0565,0566,0571,0572,      00009200
0573,0574,0575,0576,0577,0578,0579,0580,0581,0582,      00009300
0583,0584,0585,0587,0588,0589,0590,0591,0592,0593,      00009400
0594,0595,0603,0605,0606,0607,0609,0610,0611,0613,      00009500
0618,0619,0621,0622,0623,0625,0626,0627,0629,0630,      00009600
0631,0633,0634,0635,0671,0672,0673,0674,0675,0676,      00009700
0680,0681,0682,0683,0684,0685,0686,0687,0689,0690,      00009800
0691,0692,0701,0702,0703,0705,0706,0707,0709,0710,1711,      00009900
2711,3711,0713,0714,0715,0717,0718,0719,0725,0726,0727,      00010000
0729,0730,0731,0733,0734,0735,0737,0738,0739,0741,0742,      00010100
0743,0745,0746,0747,0749,0750,0751,0753,0754,0755,0771,      00010200
0772,0773,0774,0775,0776,0777,0778,0779,0783,0784,0785,      00010300
0786,0787,0788,0789,0790,0791,0792,0793,0794,0795,      00010400
0796,1803,2803,0805,0806,0807,0809,0810,0811,0813,      00010500
0814,0815,0817,0818,0819,0821,0822,0827,0829,0830,      00010600
0831,0833,0834,0835,0837,0838,0839,0841,0842,0843,      00010700
0845,0846,0847,0849,0850,0851,0853,0854,0855,0856,      00010800
0859,0861,0862,0863,0865,0866,0867,0869,0871,0872,      00010900
0873,0874,0875,0876,0877,0878,0879,0880,0881,0882,      00011000
0883,0884,0885,0886,0890,0891,0892,0893,0894,0895,      00011100
0896,0897,0898,0899,0981,0982,0903,0905,0906,0907,0992,      00011200
0993,1302,2302,3302,0338,0547,0567,1607,2607,3607,0614) ;      00011300
ON ENDFILE(TPI) EOF = YES ;      00011400
ON ENDPAGE(SYSPRINT) BEGIN ;      00011500
    PUT SKIP(2) EDIT(TIT)(X(20),A);      00011600
    END ;      00011700
EOF = NO ;      00011800
    OPEN FILE(TPI), FILE(SYSPRINT) LINESIZE(132) PAGESIZE(66);      00011900
    PUT SKIP(2) EDIT(TIT)(X(20),A);      00012000
    I=I+1;      00012100
    READ FILE(TPI) INTO(TTI(I));      00012200
    IF SUBSTR(TTI(I),19,2) ^= '00' THEN DO ;      00012300
        I=I+1;      00012400
    ;      00012500
    ;      00012600

```

```

TTI(2)=TTI(1); 00012700
TTI(1)=''; 00012800
END; 00012900
SKEY = SUBSTR(TTI(1),7,12); 00013000
DO WHILE(EOF=NO); 00013100
CALL PROCESS_R; 00013200
END; 00013300
CLOSE FILE(TPI); 00013400
PROCESS_R: PROC; 00013500
IF SKEY ^= SUBSTR(TTI(1),7,12) THEN 00013600
DO; I=I-1; 00013700
CALL EDIT_R; 00013800
IF SUBSTR(TTI(I+1),19,2)='00' 00013900
THEN DO; TTI(1)=TTI(I+1); 00014000
I=1; 00014100
END; 00014200
ELSE DO; TTI(2)=TTI(I+1); 00014300
I=2; 00014400
END; 00014500
SKEY = SUBSTR(TTI(1),7,12); 00014600
END; 00014700
I=I+1; 00014800
READ FILE(TPI) INTO(TTI(I)); 00014900
END PROCESS_R; 00015000
EDIT_R: PROC; 00015100
TII=TTI(1); 00015200
CALL EDIT_GAGU; 00015300
DO J = 2 TO I; 00015400
TCC= TTI(J); 00015500
CALL EDIT_GAGUWON; 00015600
END; 00015700
IF ESW = '0' THEN CALL RELATIVE_EDIT; 00015899
ESW = '0'; 00015900
TII=''; 00016000
TCC=''; 00016100
ERR=(100)''; 00016200
END EDIT_R; 00016300
EDIT_GAGU: PROC; 00016400
IF TTI(1)='' THEN DO; 00016500
PUT SKIP EDIT(SKEY,'NO GAGU SAHANG') (X(5),A,A); 00016600
SW='1'; 00016700
END; 00016800
IF CARD ^= '1' THEN DO; 00016900
SW = '1'; 00017000
SUBSTR(ERR,1,1) = ASK; 00017100
END; 00017200
DO II=1 BY 1 UNTIL(JOSA=DJOSA(II)|II>572);END; 00017300
IF II>572 THEN DO; 00017400
SW='1'; 00017500
SUBSTR(ERR,7,4) = ASK; 00017600
END; 00017700
IF SUBSTR(GAND,1,4) < '0101' | 00017800
SUBSTR(GAND,1,4) > '2499' THEN DO; 00017900
SW = '1'; 00018000
SUBSTR(ERR,11,4) = ASK; 00018100
END; 00018200
IF SUBSTR(GAND,5,2) < '01' | 00018300
SUBSTR(GAND,5,2) > '30' THEN DO; 00018400
SW = '1'; 00018500
SUBSTR(ERR,15,2) = ASK; 00018600
END; 00018700
IF SUBSTR(GAND,7,2) < '01' | 00018800
SUBSTR(GAND,7,2) > '30' THEN DO; 00018900

```

```

SW = '1';
SUBSTR(ERR,17,2) = ASK;
END;
IF GU < '1' | GU > '5' THEN DO;
SW = '1';
SUBSTR(ERR,21,1) = ASK;
END;
IF SOU < '1' | SOU > '6' THEN DO;
SW = '1';
SUBSTR(ERR,22,1) = ASK;
END;
IF GUK = '1' THEN GO TO BB1;
IF GUK = '2' THEN GO TO SBS;
SW = '1';
SUBSTR(ERR,23,2) = ASK;
GO TO BB2;
SBS: IF GAG > '0' THEN DO;
SW = '1';
SUBSTR(ERR,23,2) = ASK;
END;
GO TO BB2;
BB1: IF GAG < '1' | GAG > '4' THEN DO;
SW = '1';
SUBSTR(ERR,21,2) = ASK;
END;
BB2: IF GAG = '4' & (SAYU < '01' | SAYU > '15') THEN DO ;
SW = '1';
SUBSTR(ERR,25,2) = ASK;
END;
IF SW = '1' THEN DO ;
PUT SKIP(3) EDIT(ERR)(X(20),A);
PUT SKIP(1) EDIT(CARD,ILNO,JOSA,GANO,GU,SOU,GUK,GAG,SAYU)
(X(20),A,A,A,A,A,A,A,A);
CALL HPUT1('OS',0,46,30,ADDR) ;
ESW = '1';
END ;
ERR = (100) ' ';
SW = '0';
END EDIT_GAGU ;
EDIT_GAGUWON : PROC ;
IF INO < '01' | INO > '40' THEN DO;
SW = '1';
SUBSTR(ERR,19,2) = ASK;
END;
IF GAJU1 < '00' | GAJU1 > '09' THEN DO;
IF GAJU1 ^= '99'
THEN DO ;
SW = '1';
SUBSTR(ERR,41,4) = ASK;
END;
END;
IF GAJU2 < '00' | GAJU2 > '40'
THEN DO ;
IF GAJU2 ^= '99' & GAJU2 ^= '88'
THEN DO;
SW = '1';
SUBSTR(ERR,41,4) = ASK;
END;
END;
IF SEX ^= '1' & SEX ^= '2'
THEN DO;
SW = '1';
SUBSTR(ERR,45,1) = ASK;
END;

```

```

00019000
00019100
00019200
00019300
00019400
00019500
00019600
00019700
00019800
00019900
00020000
00020100
00020200
00020300
00020400
00020500
00020600
00020700
00020800
00020900
00021000
00021100
00021200
00021300
00021400
00021500
00021600
00021700
00021800
00021900
00022000
00022100
00022200
00022300
00022400
00022500
00022600
00022700
00022800
00022900
00023000
00023100
00023200
00023300
00023400
00023500
00023600
00023700
00023800
00023900
00024000
00024100
00024200
00024300
00024400
00024500
00024600
00024700
00024800
00024900
00025000
00025100
00025200

```

```

IF BIRY < 880 | BIRY > 985 THEN DO ;                                00025300
    SW = '1';                                                    00025400
    SUBSTR(ERR,46,3) = ASK;                                       00025500
    END;                                                            00025600
IF BIRM < 01 | BIRM > 12 THEN DO ;                                00025700
    SW = '1';                                                    00025800
    SUBSTR(ERR,49,2) = ASK;                                       00025900
    END;                                                            00026000
IF BIRD < 01 | BIRD > 31 THEN DO ;                                00026100
    SW = '1';                                                    00026200
    SUBSTR(ERR,51,2) = ASK;                                       00026300
    END;                                                            00026400
IF HAK1 < '00' | HAK1 > '18' THEN DO;                             00026500
    IF HAK1 ^= '99' THEN DO;                                       00026600
        SW = '1';                                                 00026700
        SUBSTR(ERR,53,3) = ASK;                                     00026800
        END;                                                       00026900
    END;                                                            00027000
IF HAK2 < '0' | HAK2 > '3' THEN DO;                               00027100
    IF HAK2 ^= '9' THEN DO;                                       00027200
        SW = '1';                                                 00027300
        SUBSTR(ERR,53,3) = ASK;                                     00027400
        END;                                                       00027500
    END;                                                            00027600
IF (BAWO < '1' | BAWO > '4') THEN DO;                             00027700
    IF BAWO ^= '9' THEN DO;                                       00027800
        SW = '1';                                                 00027900
        SUBSTR(ERR,56,1) = ASK;                                     00028000
        END;                                                       00028100
    END;                                                            00028200
IF CHY > 85 THEN DO ;                                           00028300
    SW = '1';                                                    00028400
    SUBSTR(ERR,57,2) = ASK;                                       00028500
    END;                                                            00028600
IF CHM < 00 | CHM > 12 THEN DO ;                                  00028700
    SW = '1';                                                    00028800
    SUBSTR(ERR,59,2) = ASK;                                       00028900
    END;                                                            00029000
IF CHN < '0' | CHN > '22' THEN DO ;                                00029100
    IF CHN ^= ' ' THEN DO ;                                         00029200
        IF CHN ^= '99' THEN DO ;                                     00029300
            SW = '1';                                             00029400
            SUBSTR(ERR,61,2) = ASK;                                 00029500
        END;                                                       00029600
    END;                                                            00029700
    END;                                                            00029800
/* IF (SEX='1' | BAWO='4') & IMSN||IMSY||IMSM||IMSR ^= ''        00029900
    THEN DO; SW='1';                                               00030000
    SUBSTR(ERR,61,7)=ASK; */                                       00030100
IF (SEX='1' | BAWO='4'|GAJU1='03'                                  00030200
    |GAJU1='04') THEN GOTO IMBYPASS;                                00030300
/* IF IMSY < '000' | IMSY > '985' THEN DO ;                       00030400
    SW = '1';                                                    00030500
    SUBSTR(ERR,63,4) = ASK;                                       00030600
    END;                                                            00030700
IF IMSM < '00' | IMSM > '12' THEN DO ;                             00030800
    SW = '1';                                                    00030900
    SUBSTR(ERR,63,4) = ASK;                                       00031000
    END;                                                            00031100
IF IMSR < '0' | IMSR > '4' THEN DO ;                               00031200
    SW = '1';                                                    00031300
    SUBSTR(ERR,67,1) = ASK;                                       00031400
    END;                                                            00031500
*/

```

```

IMBYPASS:
IF JUNY < '000' | JUNY > '985' THEN DO ;
  IF JUNY = ' ' THEN JUNY = '985' ;
  ELSE DO ;
    SW = '1' ;
    SUBSTR(ERR,68,3) = ASK ;
    END ;
  END ;
IF JUNM < '00' | JUNM > '12' THEN DO ;
  IF JUNM = ' ' THEN JUNM = '02' ;
  ELSE DO ;
    SW = '1' ;
    SUBSTR(ERR,70,2) = ASK ;
    END ;
  END ;
IF JUND < '00' | JUND > '31' THEN DO ;
  IF JUND = ' ' THEN JUND = '01' ;
  ELSE DO ;
    SW = '1' ;
    SUBSTR(ERR,72,2) = ASK ;
    END ;
  END ;

IF SW = '1' THEN DO ;
  PUT SKIP(3) EDIT(ERR)(X(20),A) ;
  PUT SKIP(1) EDIT(CARD1,IND,(STRING(GITA1)))
    (X(20),A,A,X(20),A) ;
  CALL HPUT1('0S',0,45,10,NAME) ;
  ESW='1' ;
  END ;
  ERR = ' ' ;
  SW = '0' ;
END EDIT_GAGUWON ;
RELATIVE_EDIT : PROC ;
  DCL ER(15) CHAR(100) INIT((15)''') ;
  DCL 1 AGE_WORK,
    2 AFROM,
      3 FY PIC '999',
      3 FM PIC '99',
      3 FD PIC '99' ,
    2 ATO,
      3 TY PIC '999',
      3 TM PIC '99',
      3 TD PIC '99',
    2 ACAL,
      3 CY PIC '99',
      3 CM PIC '99' ,
      3 CD PIC '99' ;
  SW='0' ;
  DO J=2 BY 1 UNTIL (J>I | SUBSTR(TTI(J),41,2)='01') ; END ;
  IF J>I THEN DO ;
    SW='1' ;
    PUT SKIP EDIT('NO GAGUJU =',SKEY)
      (X(20),A,A) ;
    END ;
DO J = 2 TO I ;
  AFROM = SUBSTR(TTI(J),46,7) ;
  TY = 984 ;
  TM = 12 ;
  TD = 31 ;
  IF FD > TD THEN DO ;
    TM = TM -1 ;
    CD = TD+30-FD ; END ;
  ELSE CD = TD - FD ;
  IF FM > TM THEN DO ;
    TY = TY -1 ;
    CM = TM+12-FM ; END ;

```

```

00031600
00031700
00031800
00031900
00032000
00032100
00032200
00032300
00032400
00032500
00032600
00032700
00032800
00032900
00033000
00033100
00033200
00033300
00033400
00033500
00033600
00033700
00033800
00033900
00034000
00034100
00034200
00034300
00034400
00034599
00034600
00034700
00034800
00034900
00035000
00035100
00035200
00035300
00035400
00035500
00035600
00035700
00035800
00035900
00036000
00036100
00036200
00036300
00036400
00036500
00036600
00036700
00036800
00036900
00037000
00037100
00037200
00037300
00037400
00037500
00037600
00037700
00037800

```

ELSE CH = TM - FM ;	00037900
CY = TY - FY ;	00038000
PUT DATA(CY) ;	00038100
IF CY < 15 & SUBSTR(TTI(J),57,4)=""	00038200
THEN DO ;	00038300
SW='1' ;	00038400
SUBSTR(ER(J),46,7) = ASK ;	00038500
SUBSTR(ER(J),57,4) = ASK ;	00038600
END ;	00038700
IF CY < 15 & SUBSTR(ER(J),43,2) = '00'	00038800
THEN DO ;	00038900
SW='1' ;	00039000
SUBSTR(ER(J),43,2) = ASK ;	00039100
SUBSTR(ER(J),46,7) = ASK ;	00039200
END ;	00039300
IF (CY > 6   CY < 15) &	00039400
(SUBSTR(TTI(J),53,2) < '01'	00039500
SUBSTR(TTI(J),53,2) > '06') THEN DO ;	00039600
SW='1' ;	00039700
SUBSTR(ER(J),53,3)=ASK ;	00039800
END ;	00039900
IF CY < 12 & SUBSTR(TTI(J),55,1) = '2'	00040000
THEN DO ;	00040100
SW='1' ;	00040200
SUBSTR(ER(J),53,3)=ASK ;	00040300
END ;	00040400
IF (CY > 12   CY < 17) &	00040500
(SUBSTR(TTI(J),53,2) < '07'	00040600
SUBSTR(TTI(J),53,2) > '09') THEN DO ;	00040700
SW='1' ;	00040800
SUBSTR(ER(J),53,2)=ASK ;	00040900
SUBSTR(ER(J),57,4)=ASK ;	00041000
END ;	00041100
IF CY < 15 & SUBSTR(TTI(J),55,1) = '2'	00041200
THEN DO ;	00041300
SW='1' ;	00041400
SUBSTR(ER(J),53,3)=ASK ;	00041500
END ;	00041600
IF (CY > 15   CY < 20) &	00041700
(SUBSTR(TTI(J),53,2) < '10'	00041800
SUBSTR(TTI(J),53,2) > '12') THEN DO ;	00041900
SW='1' ;	00042000
SUBSTR(ER(J),53,2)=ASK ;	00042100
SUBSTR(ER(J),57,4)=ASK ;	00042200
END ;	00042300
IF CY < 18 & SUBSTR(TTI(J),55,1) = '2'	00042400
THEN DO ;	00042500
SW='1' ;	00042600
SUBSTR(ER(J),53,3)=ASK ;	00042700
END ;	00042800
IF CY < 18 &	00042900
(SUBSTR(TTI(J),53,2) < '13'	00043000
SUBSTR(TTI(J),53,2) > '14') THEN DO ;	00043100
SW='1' ;	00043200
SUBSTR(ER(J),53,3)=ASK ;	00043300
END ;	00043400
IF CY < 20 & SUBSTR(TTI(J),55,1) = '2'	00043500
THEN DO ;	00043600
SW='1' ;	00043700
SUBSTR(ER(J),53,3)=ASK ;	00043800
END ;	00043900
IF CY < 20 & SUBSTR(TTI(J),53,2) < '15'	00044000
THEN DO ;	00044100

```

                SW='1' ;
                SUBSTR(ER(J),53,2)=ASK;
                END;
            IF CY < 22 & SUBSTR(TTI(J),55,1)='2'
            THEN DO ;
                SW='1' ;
                SUBSTR(ER(J),53,3)=ASK;
                END;
            IF CY > 22 & SUBSTR(TTI(J),53,2) < '15'
            THEN DO;
                SW='1' ;
                SUBSTR(ER(J),53,3)=ASK;
                END;
            IF CY < 24 & SUBSTR(TTI(J),55,1)='2'
            THEN DO ;
                SW='1' ;
                SUBSTR(ER(J),53,3)=ASK;
                END;
            IF SUBSTR(TTI(J),41,2) = '02' & SUBSTR(TTI(J),56,1) = '4'
            THEN DO;
                SW='1' ;
                SUBSTR(ER(J),41,2)=ASK;
                SUBSTR(ER(J),56,1)=ASK;
                END ;
            IF (SUBSTR(TTI(J),41,2) = '03' | SUBSTR(TTI(J),41,2) = '04')
            & (SUBSTR(TTI(J),56,1) = '4')
            THEN DO;
                SW='1' ;
                SUBSTR(ER(J),41,2)=ASK;
                SUBSTR(ER(J),56,1)=ASK;
                END ;
            IF (SUBSTR(TTI(J),41,2) = '05' | SUBSTR(TTI(J),41,2) = '06')
            & (SUBSTR(TTI(J),56,1) < '1' | SUBSTR(TTI(J),56,1) > '3')
            THEN DO;
                SW='1' ;
                SUBSTR(ER(J),41,2)=ASK;
                SUBSTR(ER(J),56,1)=ASK;
                END ;
            IF (SUBSTR(TTI(J),56,1) = '1' |
            SUBSTR(TTI(J),56,1) = '2' |
            SUBSTR(TTI(J),56,1) = '3' ) & SUBSTR(TTI(J),57,4)=''
            THEN DO ;
                SW='1' ;
                SUBSTR(ER(J),56,1) = ASK;
                SUBSTR(ER(J),57,4)= ASK;
                END ;
            IF SUBSTR(TTI(J),45,1) = '1' THEN DO ;
                IF SUBSTR(TTI(J),61,7)=""
                THEN DO ;
                    SW='1' ;
                    SUBSTR(ER(J),45,1) = ASK;
                    SUBSTR(ER(J),61,7) = ASK;
                    END ;
                END ;
            END ;
        IF SW = '1' THEN
            DO J=1 TO I;
                PUT SKIP(3) EDIT(ER(J))(X(20),A);
                PUT SKIP(1) EDIT(SUBSTR(TTI(J),1,20),SUBSTR(TTI(J),41,60))
                (X(20),A,X(20),A);
                CALL HPUT1('05',0,45,10,SUBSTR(TTI(J),21,20));
                END ;
            ER(*) = (100) ' ' ;
        END RELATIVE_EDIT ;
        %INCLUDE HANPRTS ;
        END RIVER;

```

```

00044200
00044300
00044400
00044500
00044600
00044700
00044800
00044900
00045000
00045100
00045200
00045300
00045400
00045500
00045600
00045700
00045800
00045900
00046000
00046100
00046200
00046300
00046400
00046500
00046600
00046700
00046800
00046900
00047000
00047100
00047200
00047300
00047400
00047500
00047600
00047700
00047800
00047900
00048000
00048100
00048200
00048300
00048400
00048500
00048600
00048700
00048800
00048900
00049000
00049100
00049200
00049300
00049400
00049500
00049600
00049700
00049800
00049900
00050000
00050100
00050299
00050300
00050400

```



```
/*
//GO.TPI DD DSN=A33.VH.TEM(SVHS1093),UNIT=DISK,VOL=SER=BOS005,
//          DCB=(RECFM=FB,LRECL=100,BLKSIZE=4000),DISP=OLD
//GO.SYSPRINT DD SYSOUT=*
//
```

```
00050500
00050600
00050700
00050800
00050900
```

A33,86.147,11:29:52,A33.PROGRAM.LIBRARY(SVHWCA52)

```
//A33VHWCA JOB CLASS=A,MSGCLASS=X,NOTIFY=A33 00000100
//STEP1 EXEC PLIFCLG 00000200
//PLI.SYSIN DD * 00000300
* PROCESS S,GS,A,XREF,OPT(2); 00000400
  LEE: PROC OPTIONS(MAIN); 00000500
    DCL MAS FILE RECORD INPUT ; 00000600
    DCL NEW FILE RECORD OUTPUT ; 00000700
    DCL TA CHAR(100); 00000800
    DCL I IN DEF TA, 00000900
      2 BB CHAR(6), 00001000
      2 GI CHAR(14), 00001100
      2 NAME CHAR(20), 00001200
      2 FF CHAR(4), 00001300
      2 SE CHAR(1), 00001400
      2 BIR CHAR(7), 00001500
      2 HAK CHAR(3), 00001600
      2 EE CHAR(1), 00001700
      2 GG CHAR(4), 00001800
      2 CHL CHAR(39), 00001900
      2 BBB CHAR(1); 00002000
    DCL MM CHAR(100); 00002100
    ON ENDFILE(MAS) GO TO CLOS; 00002200
    ON ENDFILE(SYSIN) GO TO PASS; 00002300
    OPEN FILE(MAS), FILE(NEW), FILE(SYSPRINT) LINESIZE(132); 00002400
RD: GET SKIP EDIT(MM)(A(100)); 00002500
    SUBSTR(TA,I,100) = MM; 00002600
    WRITE FILE(NEW) FROM(TA); 00002700
    GO TO RD; 00002800
  PASS: READ FILE(MAS) INTO(TA); 00002900
  /* ***** CHANGE ***** */ 00003000
  IF SUBSTR(GI,1,4) = '0671' THEN GO TO PASS; 00003100
  /* IF GI = '031305030601' & NO = '02' THEN FF = '0200' ; */ 00003200
  /* IF GI = '043304060101' & NO = '02' THEN BIR= '9560520' ; */ 00003300
    WRITE FILE(NEW) FROM(TA); 00003400
    GO TO PASS; 00003500
  CLOS: CLOSE FILE(MAS), FILE(NEW); 00003600
END LEE; 00003700
/* 00003800
//GO.SYSIN DD * 00003900
/* 00004000
//GO.MAS DD DSN=A33.VH.S10,UNIT=DISK,VOL=SER=BOS005, 00004100
// DISP=(OLD,KEEP),DCB=(RECFM=FB,BLKSIZE=4000,LRECL=100) 00004200
//GO.NEW DD DSN=A33.VH.TEM(SVHB1093),UNIT=DISK,VOL=SER=BOS005, 00004300
// DCB=(RECFM=FB,BLKSIZE=4000,LRECL=100),DISP=(OLD,KEEP) 00004400
/** SPACE=(3200,(330,5)) 00004500
//GO.SYSPRINT DD SYSOUT=* 00004600
//GO.PR DD SYSOUT=* 00004700
// 00004800
```

A33.86.147,11:30:43,A33.PROGRAM.LIBRARY(SVHWDV93)

```
//A33VHWDV JOB CLASS=A,MSGCLASS=X,NOTIFY=A33 SVH00010
//BACKUP01 EXEC PLIFCLG SVH00020
//PLI.SYSIN DD * SVH00030
* PROCESS S,GS,A,XREF,OPT(2); SVH00040
  LEE: PROC OPTIONS(MAIN); SVH00050
    DCL MAS FILE RECORD INPUT; SVH00060
    DCL NEW FILE RECORD OUTPUT; SVH00070
    DCL TA CHAR(80); SVH00080
    DCL I IN DEF TA, SVH00090
      2 BB CHAR(6), SVH00100
      2 GI CHAR(12), SVH00110
      2 SS CHAR(2), SVH00120
      2 S2 CHAR(1), SVH00130
      2 S3 CHAR(1), SVH00140
      2 NO CHAR(2), SVH00150
      2 FF CHAR(2), SVH00160
      2 PP CHAR(2), SVH00170
      2 SE CHAR(1), SVH00180
      2 BIR CHAR(7), SVH00190
      2 HAK PIC'(2)9', SVH00200
      2 EE CHAR(1), SVH00210
      2 CHA CHAR(4), SVH00220
      2 CHL CHAR(36), SVH00230
      2 BBB CHAR(1), SVH00240
    DCL MM CHAR(80); SVH00250
    DCL HAKK PIC'(2)9' INIT(0); SVH00260
    DCL CNT PIC'(6)9' INIT(0); SVH00270
    DCL (HA1,HA2) PIC'(2)9' INIT(0); SVH00280
    DCL (HB1,HB2) PIC'(2)9' INIT(0); SVH00290
    DCL (GI1,GI2) CHAR(10) INIT('0'); SVH00300
    DCL (BIR1,BIR2) CHAR(7) INIT('0'); SVH00310
    DCL (NO1,NO2) CHAR(2) INIT('0'); SVH00320
    ON ENDFILE(MAS) GO TO CLOS; SVH00330
    ON ENDFILE(SYSIN) GO TO PASS; SVH00340
    OPEN FILE(MAS), FILE(NEW), FILE(SYSPRINT) LINESIZE(132); SVH00350
  RD: GET SKIP EDIT(MM)(A(80)); SVH00360
      SUBSTR(TA,1,80) = MM; SVH00370
      WRITE FILE(NEW) FROM(TA); SVH00380
      GO TO RD; SVH00390
  PASS: READ FILE(MAS) INTO(TA); SVH00400
      WRITE FILE(NEW) FROM(TA); SVH00410
      CNT = CNT + 1; SVH00420
      GO TO PASS; SVH00430
  CLOS: CLOSE FILE(MAS), FILE(NEW); SVH00440
      PUT SKIP(5) EDIT(' BACK-UP CNT = ',CNT)(A,F(6)); SVH00450
  END LEE; SVH00460
/* SVH00470
//GO.SYSIN DD * SVH00480
/* SVH00490
//GO.MAS DD DSN=SVHTAPE,UNIT=TAPE,VOL=SER=TN6272,DISP=(OLD,KEEP), SVH00520
// DCB=(RECFM=FB,BLKSIZE=3200,LRECL=80),LABEL=(2,BLP) SVH00530
//GO.NEW DD DSN=A33.VH.KK(SELECT),UNIT=DISK,VOL=SER=BOS005, SVH00500
// DCB=(RECFM=FB,BLKSIZE=3200,LRECL=80),DISP=(OLD,KEEP) SVH00510
//GO.SYSPRINT DD SYSOUT=* SVH00540
//GO.PR DD SYSOUT=* SVH00550
// SVH00560
```

A33,86.055,18:45:14,A33.PROGRAM.LIBRARY(GAMERGE)

```
//A33VHUM1 JOB TYPRUN=HOLD,CLASS=A,MSGCLASS=X 00000100
//GAMERGE EXEC PGM=SORT,PARM='SIZE(MAX)' 00000200
//SYSOUT DD SYSOUT=A 00000300
//SYSPRINT DD SYSOUT=A 00000400
//SORTIN DD DSN=SVHTAPE,UNIT=TAPE,DISP=(OLD,KEEP),LABEL=(2,BLP), 00000500
// DCB=(RECFM=FB,BLKSIZE=4000,LRECL=100),VOL=SER=TN4867 00000600
// DD DSN=A33.VH.TEM(SVHG0693),UNIT=DISK,DISP=(OLD,KEEP), 00000700
// DCB=(RECFM=FB,BLKSIZE=4000,LRECL=100),VOL=SER=BOS005 00000800
//SORTOUT DD DSN=SVHTBBE,UNIT=TAPE,DISP=(NEW,KEEP), 00000900
// DCB=(RECFM=FB,BLKSIZE=4000,LRECL=100),VOL=SER=SCRTCH 00001000
//SORTWK01 DD UNIT=DISK,VOL=SER=SORTWK,SPACE=(CYL,(100)) 00001100
//SYSIN DD * 00001200
// SORT FIELDS=(7,14,A),FORMAT=CH 00001300
/* 00001400
// 00001500
```

A33,86.055,17:26:23,A33.PROGRAM.LIBRARY(BUSORT)

```
//H01VHUSB JOB CLASS=A,MSGCLASS=X,NOTIFY=H01, TYPRUN=HOLD 00000100
//SORT EXEC PGM=SORT,PARM='SIZE(MAX)' 00000200
//SYSOUT DD SYSOUT=* 00000300
//SYSPRINT DD SYSOUT=* 00000400
//SORTIN DD DSN=SVHTAPE,UNIT=TAPE,DISP=(OLD,KEEP),LABEL=(,NL), 00000500
// DCB=(RECFM=F,BLKSIZE=100,LRECL=100,EROPT=SKP), 00000600
// VOL=SER=TN6801 00000700
//SORTOUT DD DSN=A33.VH.TEM(SVHB0993),UNIT=DISK,DISP=(OLD,KEEP), 00000800
// VOL=SER=BOS005, 00000900
// DCB=(RECFM=FB,LRECL=100,BLKSIZE=4000) 00001000
//SORTWK01 DD UNIT=DISK,SPACE=(CYL,(100)),VOL=SER=SORTWK 00001100
//SYSIN DD * 00001200
SORT FIELDS=(10,4,A,15,8,A,14,1,A,23,4,A),FORMAT=BI,FILSZ=E60000 00001300
/* 00001400
// 00001500
```

```

//A33VHWEB JOB CLASS=B,MSGCLASS=X,NOTIFY=A33          00000100
//BUCHECK EXEC PLIFCLG                                00000200
//PLI.SYSIN DD *                                      00000300
* PROCESS S,GS,NEST,OPT(2);                            00000400
  RIVER: PROC OPTIONS(MAIN);                             00000500
    DCL TPI FILE RECORD INPUT ;                          00000600
      DCL 1 TA,                                           00000700
        2 CARD      CHAR(1),                               00000800
        2 GI        CHAR(12),                             00000900
        2 GUBUN     CHAR(1),                               00001000
        2 GITA      CHAR(86);                             00001100
      DCL 1 CA1 DEF GITA,                                  00001200
        2 GNO1      CHAR(8),                               00001300
        2 GWON1     CHAR(2),                               00001400
        2 ILJA1     CHAR(6),                               00001500
        2 JGEO      CHAR(5),                               00001600
        2 IDON      CHAR(4),                               00001700
        2 GUB1      CHAR(2),                               00001800
        2 BLK1      CHAR(59);                             00001900
      DCL 1 CA2 DEF GITA,                                  00002000
        2 GNO2      CHAR(8),                               00002100
        2 GWON2     CHAR(2),                               00002200
        2 ILJA2     CHAR(6),                               00002300
        2 MOK       CHAR(5),                               00002400
        2 IDON21    CHAR(2),                               00002500
        2 IDON22    CHAR(2),                               00002600
        2 GUB2      CHAR(2),                               00002700
        2 BLK2      CHAR(59);                             00002800
      DCL 1 CA3 DEF GITA ,                                 00002900
        2 GNO3      CHAR(8),                               00003000
        2 GWON3     CHAR(2),                               00003100
        2 NAME3     CHAR(20),                              00003200
        2 GAJU      CHAR(4),                               00003300
        2 SEX       CHAR(1),                               00003400
        2 BIRTH     CHAR(7),                               00003500
        2 HAK       CHAR(3),                               00003600
        2 BAWO     CHAR(1),                               00003700
        2 CHON     CHAR(4),                               00003800
        2 CHPN     CHAR(2),                               00003900
        2 IMSN     CHAR(5),                               00004000
        2 JGEO3    CHAR(5),                               00004100
        2 JNIB     CHAR(6),                               00004200
        2 IDON31   CHAR(2),                               00004300
        2 IDON32   CHAR(2),                               00004400
        2 GUB3     CHAR(2),                               00004500
        2 BLK3     CHAR(12);                              00004600
      DCL 1 CA4 DEF GITA ,                                 00004700
        2 GNO4      CHAR(8),                               00004800
        2 GWON4     CHAR(2),                               00004900
        2 NAME4     CHAR(20),                              00005000
        2 BYUN4    CHAR(6),                               00005100
        2 MOK4     CHAR(5),                               00005200
        2 IDON41   CHAR(2),                               00005300
        2 IDON42   CHAR(2),                               00005400
        2 GUB4     CHAR(2),                               00005500
        2 BLK4     CHAR(39);                              00005600
      DCL 1 CA5 DEF GITA ,                                 00005700
        2 GNO5      CHAR(8),                               00005800
        2 GWON5     CHAR(2),                               00005900
        2 GPYO     CHAR(2),                               00006000
        2 SUJEON   CHAR(7),                               00006100
        2 SUWHO    CHAR(7),                               00006200
        2 BLK5     CHAR(60);                              00006300

```

```

DCL 1 CA6 DEF GITA ,                                00006400
      2 GUB6 CHAR(2) ,                               00006500
      2 IDJGN CHAR(8) ,                               00006600
      2 IDJGW CHAR(2) ,                               00006700
      2 IDWGN CHAR(8) ,                               00006800
      2 IDWGW CHAR(2) ,                               00006900
      2 GAJU6 CHAR(4) ,                               00007000
      2 GEO CHAR(1) ,                                00007100
      2 SOU CHAR(1) ,                                00007200
      2 BAW CHAR(1) ,                                00007300
      2 CHYN CHAR(4) ,                               00007400
      2 BLK6 CHAR(53) ;                               00007500
DCL 1 CA7 DEF GITA ,                                00007600
      2 GAND7 CHAR(8) ,                               00007700
      2 GWON7 CHAR(2) ,                               00007800
      2 NAME7 CHAR(20) ,                              00007900
      2 CHPLN CHAR(4) ,                               00008000
      2 CHR CHAR(1) ,                                00008100
      2 BLK7 CHAR(51) ;                               00008200
DCL BNT PIC(4)9' INIT(0) ;                           00008300
DCL SWE CHAR(1) INIT('0') ;                          00008400
DCL ANT PIC(4)9' INIT(0) ;                           00008500
DCL ASK CHAR(20) INIT((20)'*') ;                     00008600
DCL SIDON(31) CHAR(2) INIT('00','11','12','13','14', 00008700
      '15','19','20','31','32','39','41','42', 00008800
      '43','44','49','51','52','59','61','62', 00008900
      '69','71','72','73','79','81','82','89', 00009000
      '93','99') ; 00009100
DCL IGUB(5) CHAR(2) INIT('11','21','22','91','93') ; 00009200
DCL DGUB(6) CHAR(2) INIT('51','61','62','75','92','94') ; 00009300
DCL IWGUB(4) CHAR(2) INIT('12','21','22','25') ;     00009400
DCL DWGUB(4) CHAR(2) INIT('52','61','62','75') ;     00009500
DCL JGUB(4) CHAR(2) INIT('01','02','03','04') ;      00009600
DCL TTT CHAR(90) INIT('.....5...10...15...20... 00009700
25...30...35...40...45...50...55...60...65...70...75...80...85...90') ; 00009800
DCL ERR CHAR(90) INIT(' ');                           00009900
OPEN FILE(TPI), FILE(SYSPRINT) LINESIZE(132);        00010000
ON ENDPAGE BEGIN ;                                   00010100
PUT SKIP(3) EDIT(TTT)(X(18),A);                       00010200
END ;                                                  00010300
ON ENDFILE(TPI) GO TO LAST;                           00010400
PUT SKIP(3) EDIT(TTT)(X(18),A);                       00010500
RD: READ FILE(TPI) INTO (TA);                          00010600
ANT = ANT + 1;                                         00010700
IF CARD = '2' THEN GO TO AA;                           00010800
SUBSTR(ERR,1,1)='*';                                   00010900
AA: IF GUBUN = '1' THEN GO TO CARD1;                   00011000
IF GUBUN = '2' THEN GO TO CARD2;                       00011100
IF GUBUN = '3' THEN GO TO CARD3;                       00011200
IF GUBUN = '4' THEN GO TO CARD4;                       00011300
IF GUBUN = '5' THEN GO TO CARD5;                       00011400
IF GUBUN = '6' THEN GO TO CARD6;                       00011500
IF GUBUN = '7' THEN GO TO CARD7;                       00011600
PUT SKIP(2) EDIT('NOTHING CARD',CARD,GI,GUBUN,GITA) 00011700
      (X(18),A(12),X(2),4 A); 00011800
GO TO RD;                                              00011900
CARD1: IF GNO1 < '01010101' |                          00012000
      GNO1 > '24993030' THEN DO;                       00012100
      SWE = '1';                                         00012200
      SUBSTR(ERR,15,8) = ASK;                            00012300
      END;                                               00012400
IF GWON1 < '01' |                                     00012500

```

```

        GWON1 > '45'                THEN DO;                00012700
                                SWE = '1';                00012800
                                SUBSTR(ERR,23,2) = ASK;    00012900
                                END;                    00013000
IF GUB1 = '22' THEN
DO, IF JGEO '=' | IDON '=' THEN
    DO ; SWE='1';                00013100
        SUBSTR(ERR,40,2)=ASK ;    00013200
    END ;                            00013300
ELSE DO ;                            00013400
    IF SUBSTR(JGEO,1,1) < '2' |      00013500
        SUBSTR(JGEO,1,1) > '5' & SUBSTR(JGEO,1,1) < '9' THEN 00013600
    DO ; SWE = '1';                00013700
        SUBSTR(ERR,31,1) = ASK;    00013800
    END;                            00013900
    DO II=2 BY 1 UNTIL(SUBSTR(IDON,1,2)=SIDON(II)| II>31);END; 00014000
    IF II>31 THEN DO ;              00014100
        SWE='1';                    00014200
        SUBSTR(ERR,36,2)=ASK;        00014300
        END;                            00014400
    DO II=1 BY 1 UNTIL(SUBSTR(IDON,3,2)=SIDON(II)| II>31);END; 00014500
    IF II>31 THEN DO ;              00014600
        SWE='1';                    00014700
        SUBSTR(ERR,38,2)=ASK;        00014800
        END;                            00014900
    END ;                            00015000
    IF ILJA1 < '800101' |           00015100
        ILJA1 > '851231'           00015200
                                THEN DO;                00015300
                                SWE = '1';                00015400
                                SUBSTR(ERR,25,6) = ASK;    00015500
                                END;                    00015600
    DO II=1 BY 1 UNTIL(GUB1=IGUB(II) | II >5); END ;        00015700
    IF II>5 THEN DO;                00015800
        SWE='1';                    00015900
        SUBSTR(ERR,40,2)=ASK;        00016000
        END;                            00016100
    GO TO BACK;                        00016200
CARD2:                                00016300
    IF GNO2 < '01010101' |         00016400
        GNO2 > '24993030'         00016500
                                THEN DO;                00016600
                                SWE = '1';                00016700
                                SUBSTR(ERR,15,8) = ASK;    00016800
                                END;                    00016900
    IF GWON2 < '01' |              00017000
        GWON2 > '45'                THEN DO;                00017100
                                SWE = '1';                00017200
                                SUBSTR(ERR,23,2) = ASK;    00017300
                                END;                    00017400
    IF ILJA2 < '800101' |           00017500
        ILJA2 > '851231'           00017600
                                THEN DO;                00017700
                                SWE = '1';                00017800
                                SUBSTR(ERR,25,6) = ASK;    00017900
                                END;                    00018000
    IF GUB2 = '62' | GUB2 = '75' THEN 00018100
    DO ;                            00018200
        IF MOK=' ' |                00018300
            IDON21=' ' | IDON22=' ' THEN DO ;            00018400
                SWE='1';                00018500
                SUBSTR(ERR,40,2) =ASK ; 00018600
                END ;                    00018700
    END ;                            00018800
ELSE DO ;                            00018900

```



```

IF SUBSTR(MOK,1,1) < '2' |
SUBSTR(MOK,1,1) > '5' & SUBSTR(MOK,1,1) < '9' THEN
DO ;
    SUBSTR(ERR,31,1) = ASK;
    SWE = '1';
END;
DO II=2 BY 1 UNTIL(IDON21=SIDON(II) | II > 31); END;
IF II > 31 THEN DO ;
    SWE='1';
    SUBSTR(ERR,36,2)=ASK;
END;
DO II=4 BY 1 UNTIL(IDON22=SIDON(II) | II > 31); END;
IF II > 31 THEN DO ;
    SWE='1';
    SUBSTR(ERR,38,2)=ASK;
END;
END ;
DO II=4 BY 1 UNTIL(GUB2=DGUB(II) | II>6);END;
IF II>6 THEN DO ;
    SWE='1';
    SUBSTR(ERR,40,2)=ASK;
END;
GO TO BACK;
CARD3:
IF GWON3 < '01' |
GWON3 > '45' THEN DO;
    SWE = '1';
    SUBSTR(ERR,23,2) = ASK;
END;
IF GAJU < '0000' |
GAJU > '1510' THEN DO ;
    SWE = '1';
    SUBSTR(ERR,22,2) = ASK;
END;
IF SEX < '1' |
SEX > '2' & SEX < '9' THEN DO;
    SWE = '1';
    SUBSTR(ERR,49,2) = ASK;
END;
IF BIRTH < '8800101' | BIRTH > '9851231' &
BIRTH < '9999999' THEN
DO ;
    SWE = '1';
    SUBSTR(ERR,50,7) = ASK;
END;
IF HAK < '000' |
HAK > '183' & HAK < '999' THEN DO;
    SWE = '1';
    SUBSTR(ERR,57,3) = ASK;
END;
IF BAWO < '1' | BAWO > '4' THEN DO ;
    SWE = '1';
    SUBSTR(ERR,60,1) = ASK;
END;
IF BAWO = '4' THEN DO ;
    IF CHON '=' |
    CHPN '=' |
    IMSN '=' THEN
DO ;
    SWE = '1';
    SUBSTR(ERR,60,1) = ASK;
END;
END ;
IF GUB3 = '22' | GUB3 = '25' THEN

```

```

00019000
00019100
00019200
00019300
00019400
00019500
00019600
00019700
00019800
00019900
00020000
00020100
00020200
00020300
00020400
00020500
00020600
00020700
00020800
00020900
00021000
00021100
00021200
00021300
00021400
00021500
00021600
00021700
00021800
00021900
00022000
00022100
00022200
00022300
00022400
00022500
00022600
00022700
00022800
00022900
00023000
00023100
00023200
00023300
00023400
00023500
00023600
00023700
00023800
00023900
00024000
00024100
00024200
00024300
00024400
00024500
00024600
00024700
00024800
00024900
00025000
00025100
00025200

```

```

DO ;
  IF JGEO3 '=' |
    IDON31 '=' |
    IDON32 '=' |
    JNIB '=' THEN
    DO ; SWE = '1' ;
      SUBSTR(ERR,87,2) = ASK ;
    END ;
  END ;
ELSE
DO ;
  DO II = 2 BY 1 UNTIL (IDON31=SIDON(II)|II>31);END;
  IF II>31 THEN DO;
    SWE='1';
    SUBSTR(ERR,83,2)=ASK;
    END;
  DO II = 1 BY 1 UNTIL (IDON32=SIDON(II)|II>31);END;
  IF II>31 THEN DO;
    SWE='1';
    SUBSTR(ERR,85,2)=ASK;
    END;
  END ;
  DO II=1 BY 1 UNTIL (GUB3=IWGUB(II)|II>4) ; END ;
  IF II>4 THEN DO ;
    SWE='1' ;
    SUBSTR(ERR,87,2) = ASK ;
  END ;
GO TO BACK;
CARD4:
  IF GWON4 < '01' |
    GWON4 > '45' THEN DO;
    SWE = '1';
    SUBSTR(ERR,23,2) = ASK;
  END ;
  IF GUB4 = '62' | GUB4 = '75' THEN
DO ;
  IF MOK4 '=' |
    IDON41 '=' | IDON42 '=' THEN
  DO; SWE='1' ;
    SUBSTR(ERR,60,2) =ASK ;
  END ;
  END ;
ELSE DO ;
  DO II = 2 BY 1 UNTIL (IDON41=SIDON(II)|II>31);END;
  IF II>31 THEN DO;
    SWE='1';
    SUBSTR(ERR,56,4)=ASK;
    END;
  DO II = 1 BY 1 UNTIL (IDON42=SIDON(II)|II>31);END;
  IF II>31 THEN DO;
    SWE='1';
    SUBSTR(ERR,56,4)=ASK;
    END;
  IF SUBSTR(MOK4,1,1) < '2' | SUBSTR(MOK4,1,1) > '5'
    & SUBSTR(MOK4,1,1) < '9' THEN
  DO ;
    SWE = '1' ;
    SUBSTR(ERR,51,1) = ASK ;
  END ;
  END;
  IF BYUN4 < '800101' | BYUN4 > '851231' THEN DO;
    SWE = '1';
    SUBSTR(ERR,45,6) = ASK;

```

```

00025300
00025400
00025500
00025600
00025700
00025800
00025900
00026000
00026100
00026200
00026300
00026400
00026500
00026600
00026700
00026800
00026900
00027000
00027100
00027200
00027300
00027400
00027500
00027600
00027700
00027800
00027900
00028000
00028100
00028200
00028300
00028400
00028500
00028600
00028700
00028800
00028900
00029000
00029100
00029200
00029300
00029400
00029500
00029600
00029700
00029800
00029900
00030000
00030100
00030200
00030300
00030400
00030500
00030600
00030700
00030800
00030900
00031000
00031100
00031200
00031300
00031400
00031500

```

```

                                END;
                                00031600
DO II = 1 BY 1 UNTIL(GUB4=DWGUB(II)|II>4); END;
                                00031700
IF II>4 THEN DO;
                                00031800
    SWE='1';
                                00031900
    SUBSTR(ERR,60,2)=ASK;
                                00032000
    END;
                                00032100
GO TO BACK;
                                00032200
CARD5:
                                00032300
IF GWON5 < '01' |
                                00032400
    GWON5 > '45' THEN DO;
                                00032500
    IF GWON5 = ' ' THEN DO;
                                00032600
        SWE = '1';
                                00032700
        SUBSTR(ERR,23,2) = ASK;
                                00032800
        END;
                                00032900
    END;
                                00033000
IF GPYO < '01' | GPYO > '22' THEN DO;
                                00033100
    SWE = '1';
                                00033200
    SUBSTR(ERR,25,2) = ASK;
                                00033300
    END;
                                00033400
IF GPYO = '11' | GPYO = '17' THEN
                                00033500
DO;
                                00033600
IF SUWHO = '' THEN DO;
                                00033700
    SWE = '1';
                                00033800
    SUBSTR(ERR,34,7) = ASK;
                                00033900
    END;
                                00034000
END;
                                00034100
ELSE DO;
                                00034200
IF SUJEON = '' THEN DO;
                                00034300
    SWE = '1';
                                00034400
    SUBSTR(ERR,27,7) = ASK;
                                00034500
    END;
                                00034600
IF SUWHO = '' THEN DO;
                                00034700
    SWE = '1';
                                00034800
    SUBSTR(ERR,34,7) = ASK;
                                00034900
    END;
                                00035000
END;
                                00035100
GO TO BACK;
                                00035200
CARD6:
                                00035300
IF GUB6 = '01' THEN DO;
                                00035400
    IF IDJGW = '' |
                                00035500
        IDWGW = '' |
                                00035600
        GAJU6 = '' |
                                00035700
        GEO = '' | SOU = '' | BAW = '' |
                                00035800
        CHYN = '' THEN DO;
                                00035900
        SWE = '1';
                                00036000
        SUBSTR(ERR,15,2) = ASK;
                                00036100
        END;
                                00036200
    END;
                                00036300
IF GUB6 < '01' | GUB6 > '04' THEN DO;
                                00036400
    SWE = '1';
                                00036500
    SUBSTR(ERR,15,2) = ASK;
                                00036600
    END;
                                00036700
GO TO BACK;
                                00036800
CARD7:
                                00036900
IF CHPLN = '' THEN DO;
                                00037000
    IF CHR = '' THEN DO;
                                00037100
        SWE = '1';
                                00037200
        SUBSTR(ERR,49,1) = ASK;
                                00037300
        END;
                                00037400
    END;
                                00037500
ELSE IF CHR = '' THEN DO;
                                00037600
    IF CHPLN = '' THEN DO;
                                00037700
        SWE = '1';
                                00037800
    END;

```

```

                                SUBSTR(ERR,45,4) = ASK;
                                END;
                                00037900
                                00038000
                                00038100
                                00038200
                                00038300
                                00038400
                                00038500
                                00038600
                                00038700
                                00038800
                                00038900
                                00039000
                                00039100
                                00039200
                                00039300
                                00039400
                                00039500
                                00039600
                                00039700
                                00039800
                                00039900
                                00040000
                                00040100
                                00040200
                                00040300
                                00040400
                                00040500
                                00040600
                                00040700
                                00040800
                                00040900
                                00041000
                                00041100
                                00041200
                                00041300
                                00041400
                                00041500
                                00041600
                                00041700
                                00041800
                                00041900
                                00042000
                                00042100
                                00042200
                                00042300
                                00042400
                                00042500
                                00042600
                                00042700
                                00042800
                                00042900
                                00043000
                                00043100
                                00043200
                                00043300
                                00043400
                                00043500
                                00043600
                                00043700
                                00043800
                                00043900
                                00044000

                                SUBSTR(ERR,45,5)=ASK;
                                END;
                                00038500
                                00038600
                                00038700
                                00038800
                                00038900
                                00039000
                                00039100
                                00039200
                                00039300
                                00039400
                                00039500
                                00039600
                                00039700
                                00039800
                                00039900
                                00040000
                                00040100
                                00040200
                                00040300
                                00040400
                                00040500
                                00040600
                                00040700
                                00040800
                                00040900
                                00041000
                                00041100
                                00041200
                                00041300
                                00041400
                                00041500
                                00041600
                                00041700
                                00041800
                                00041900
                                00042000
                                00042100
                                00042200
                                00042300
                                00042400
                                00042500
                                00042600
                                00042700
                                00042800
                                00042900
                                00043000
                                00043100
                                00043200
                                00043300
                                00043400
                                00043500
                                00043600
                                00043700
                                00043800
                                00043900
                                00044000

                                END;
                                00038100
                                00038200
                                00038300
                                00038400
                                00038500
                                00038600
                                00038700
                                00038800
                                00038900
                                00039000
                                00039100
                                00039200
                                00039300
                                00039400
                                00039500
                                00039600
                                00039700
                                00039800
                                00039900
                                00040000
                                00040100
                                00040200
                                00040300
                                00040400
                                00040500
                                00040600
                                00040700
                                00040800
                                00040900
                                00041000
                                00041100
                                00041200
                                00041300
                                00041400
                                00041500
                                00041600
                                00041700
                                00041800
                                00041900
                                00042000
                                00042100
                                00042200
                                00042300
                                00042400
                                00042500
                                00042600
                                00042700
                                00042800
                                00042900
                                00043000
                                00043100
                                00043200
                                00043300
                                00043400
                                00043500
                                00043600
                                00043700
                                00043800
                                00043900
                                00044000

                                IF CHPLN = '' & CHR = '' THEN DO ;
                                SWE = '1' ;
                                SUBSTR(ERR,45,5)=ASK;
                                END ;
                                00038100
                                00038200
                                00038300
                                00038400
                                00038500
                                00038600
                                00038700
                                00038800
                                00038900
                                00039000
                                00039100
                                00039200
                                00039300
                                00039400
                                00039500
                                00039600
                                00039700
                                00039800
                                00039900
                                00040000
                                00040100
                                00040200
                                00040300
                                00040400
                                00040500
                                00040600
                                00040700
                                00040800
                                00040900
                                00041000
                                00041100
                                00041200
                                00041300
                                00041400
                                00041500
                                00041600
                                00041700
                                00041800
                                00041900
                                00042000
                                00042100
                                00042200
                                00042300
                                00042400
                                00042500
                                00042600
                                00042700
                                00042800
                                00042900
                                00043000
                                00043100
                                00043200
                                00043300
                                00043400
                                00043500
                                00043600
                                00043700
                                00043800
                                00043900
                                00044000

                                BACK: IF SWE = '0' THEN GO TO RD;
                                IF GUBUN ^= '1' THEN GO TO PR1;
                                PUT SKIP(2) EDIT(ERR)(X(18),A);
                                PUT SKIP(1) EDIT('1 CARD',CARD,GI,GUBUN,GITA)
                                (X(10),A(6),X(2),A,A,A,A);
                                GO TO CLEAR;
                                00038600
                                00038700
                                00038800
                                00038900
                                00039000
                                00039100
                                00039200
                                00039300
                                00039400
                                00039500
                                00039600
                                00039700
                                00039800
                                00039900
                                00040000
                                00040100
                                00040200
                                00040300
                                00040400
                                00040500
                                00040600
                                00040700
                                00040800
                                00040900
                                00041000
                                00041100
                                00041200
                                00041300
                                00041400
                                00041500
                                00041600
                                00041700
                                00041800
                                00041900
                                00042000
                                00042100
                                00042200
                                00042300
                                00042400
                                00042500
                                00042600
                                00042700
                                00042800
                                00042900
                                00043000
                                00043100
                                00043200
                                00043300
                                00043400
                                00043500
                                00043600
                                00043700
                                00043800
                                00043900
                                00044000

                                PR1: IF GUBUN ^= '2' THEN GO TO PR2;
                                PUT SKIP(2) EDIT(ERR)(X(18),A);
                                PUT SKIP(1) EDIT('2 CARD',CARD,GI,GUBUN,GITA)
                                (X(10),A(6),X(2),A,A,A,A);
                                GO TO CLEAR;
                                00039200
                                00039300
                                00039400
                                00039500
                                00039600
                                00039700
                                00039800
                                00039900
                                00040000
                                00040100
                                00040200
                                00040300
                                00040400
                                00040500
                                00040600
                                00040700
                                00040800
                                00040900
                                00041000
                                00041100
                                00041200
                                00041300
                                00041400
                                00041500
                                00041600
                                00041700
                                00041800
                                00041900
                                00042000
                                00042100
                                00042200
                                00042300
                                00042400
                                00042500
                                00042600
                                00042700
                                00042800
                                00042900
                                00043000
                                00043100
                                00043200
                                00043300
                                00043400
                                00043500
                                00043600
                                00043700
                                00043800
                                00043900
                                00044000

                                PR2: IF GUBUN ^= '3' THEN GO TO PR3;
                                PUT SKIP(2) EDIT(ERR)(X(18),A);
                                PUT SKIP(2) EDIT('3 CARD',CARD,GI,GUBUN,SUBSTR(GITA,1,10)
                                ,SUBSTR(GITA,31,45))
                                (X(10),A(6),X(2),A,A,A,A,X(20),A);
                                CALL HPUT1('0S',0,42,20,SUBSTR(GITA,11,20));
                                GO TO CLEAR;
                                00039700
                                00039800
                                00039900
                                00040000
                                00040100
                                00040200
                                00040300
                                00040400
                                00040500
                                00040600
                                00040700
                                00040800
                                00040900
                                00041000
                                00041100
                                00041200
                                00041300
                                00041400
                                00041500
                                00041600
                                00041700
                                00041800
                                00041900
                                00042000
                                00042100
                                00042200
                                00042300
                                00042400
                                00042500
                                00042600
                                00042700
                                00042800
                                00042900
                                00043000
                                00043100
                                00043200
                                00043300
                                00043400
                                00043500
                                00043600
                                00043700
                                00043800
                                00043900
                                00044000

                                PR3: IF GUBUN ^= '4' THEN GO TO PR4;
                                PUT SKIP(2) EDIT (ERR) (X(18),A);
                                PUT SKIP(2) EDIT('4 CARD',CARD,GI,GUBUN,SUBSTR(GITA,1,10)
                                ,SUBSTR(GITA,31,20))
                                (X(10),A(6),X(2),A,A,A,A,X(20),A);
                                CALL HPUT1('0S',0,42,20,SUBSTR(GITA,11,20));
                                GO TO CLEAR;
                                00040700
                                00040800
                                00040900
                                00041000
                                00041100
                                00041200
                                00041300
                                00041400
                                00041500
                                00041600
                                00041700
                                00041800
                                00041900
                                00042000
                                00042100
                                00042200
                                00042300
                                00042400
                                00042500
                                00042600
                                00042700
                                00042800
                                00042900
                                00043000
                                00043100
                                00043200
                                00043300
                                00043400
                                00043500
                                00043600
                                00043700
                                00043800
                                00043900
                                00044000

                                PR4: IF GUBUN ^= '5' THEN GO TO PR5;
                                PUT SKIP(2) EDIT (ERR) (X(18),A);
                                PUT SKIP(1) EDIT('5 CARD',CARD,GI,GUBUN,GITA)
                                (X(10),A(6),X(2),A,A,A,A);
                                GO TO CLEAR;
                                00041100
                                00041200
                                00041300
                                00041400
                                00041500
                                00041600
                                00041700
                                00041800
                                00041900
                                00042000
                                00042100
                                00042200
                                00042300
                                00042400
                                00042500
                                00042600
                                00042700
                                00042800
                                00042900
                                00043000
                                00043100
                                00043200
                                00043300
                                00043400
                                00043500
                                00043600
                                00043700
                                00043800
                                00043900
                                00044000

                                PR5: IF GUBUN ^= '6' THEN GO TO PR6;
                                PUT SKIP(2) EDIT (ERR) (X(18),A);
                                PUT SKIP(1) EDIT('6 CARD',CARD,GI,GUBUN,GITA)
                                (X(10),A(6),X(2),A,A,A,A);
                                GO TO CLEAR;
                                00041600
                                00041700
                                00041800
                                00041900
                                00042000
                                00042100
                                00042200
                                00042300
                                00042400
                                00042500
                                00042600
                                00042700
                                00042800
                                00042900
                                00043000
                                00043100
                                00043200
                                00043300
                                00043400
                                00043500
                                00043600
                                00043700
                                00043800
                                00043900
                                00044000

                                PR6: PUT SKIP(2) EDIT (ERR) (X(18),A);
                                PUT SKIP(1) EDIT('7 CARD',CARD,GI,GUBUN,SUBSTR(GITA,1,10)
                                ,SUBSTR(GITA,31,5))
                                (X(10),A(6),X(2),A,A,A,A,X(20),A);
                                CALL HPUT1('0S',0,42,20,SUBSTR(GITA,11,20));
                                00042100
                                00042200
                                00042300
                                00042400
                                00042500
                                00042600
                                00042700
                                00042800
                                00042900
                                00043000
                                00043100
                                00043200
                                00043300
                                00043400
                                00043500
                                00043600
                                00043700
                                00043800
                                00043900
                                00044000

                                CLEAR:
                                ERR = ' ';
                                BNT = BNT + 1;
                                SWE = '0';
                                GO TO RD;
                                00042700
                                00042800
                                00042900
                                00043000
                                00043100
                                00043200
                                00043300
                                00043400
                                00043500
                                00043600
                                00043700
                                00043800
                                00043900
                                00044000

                                LAST: PUT SKIP(3) EDIT('TOTAL = ',ANT,'ERROR = ',BNT)
                                (X(18),A(8),A,X(5),A(8),A);
                                CLOSE FILE(TPI);
                                00043200
                                00043300
                                00043400
                                00043500
                                00043600
                                00043700
                                00043800
                                00043900
                                00044000

                                ZINCLUDE HANPRTS ;
                                END RIVER;
                                00043400
                                00043500
                                00043600
                                00043700
                                00043800
                                00043900
                                00044000

                                /*
                                //GO.TPI DD DSN=A33.VH.TEM(SVHB0993),UNIT=DISK,VOL=SER=BOS005,
                                // DCB=(RECFM=FB,LRECL=100,BLKSIZE=4000),DISP=OLD
                                //GO.SYSPRINT DD SYSOUT=*
                                //
                                00043700
                                00043800
                                00043900
                                00044000

```

A33,86.147,11:55:52,A33.PROGRAM.LIBRARY(SVHWBB33)

```
//A33VHBB JOB CLASS=A,MSGCLASS=X,TYPRUN=HOLD,NOTIFY=A33 00000100
//STEP1 EXEC PLIFCLG 00000200
//PLI.SYSIN DD * 00000300
* PROCESS S,GS,A,XREF,OPT(2); 00000400
LEE: PROC OPTIONS(MAIN); 00000500
    DCL MAS FILE RECORD INPUT; 00000600
    DCL NEW FILE RECORD OUTPUT; 00000700
    DCL TA CHAR(100); 00000800
    DCL 1 IN DEF TA, 00000900
        2 BB CHAR(6), 00001000
        2 GI CHAR(14), 00001100
        2 SS CHAR(8), 00001200
        2 AA CHAR(12), 00001300
        2 FF CHAR(18), 00001400
        2 CHL CHAR(22); 00001500
    DCL MM CHAR(100); 00001600
    ON ENDFILE(MAS) GO TO CLOS; 00001700
    ON ENDFILE(SYSIN) GO TO PASS; 00001800
    OPEN FILE(MAS), FILE(NEW), FILE(SYSPRINT) LINESIZE(132); 00001900
RD: GET SKIP EDIT(MM)(A(100)); 00002000
    SUBSTR(TA,1,100) = MM; 00002100
    WRITE FILE(NEW) FROM(TA); 00002200
    GO TO RD; 00002300
PASS: READ FILE(MAS) INTO(TA); 00002400
/* ***** CHANGE *****/ 00002500
    IF GI = '6724301010101310 ' THEN SUBSTR(SS,5,2) = '00' ; 00002600
    IF GI = '67419110101 0231' THEN SUBSTR(AA,3,2) = '22' ; 00002700
    WRITE FILE(NEW) FROM(TA); 00002800
    GO TO PASS; 00002900
CLOS: CLOSE FILE(MAS), FILE(NEW); 00003000
END LEE; 00003100
/* 00003200
//GO.SYSIN DD * 00003300
/* 00003400
//GO.MAS DD DSN=SVHB193,UNIT=DISK,VOL=SER=BOS005,DISP=(OLD,KEEP), 00003500
// DCB=(RECFM=FB,BLKSIZE=4000,LRECL=100) 00003600
//GO.NEW DD DSN=SVHB293,UNIT=DISK,VOL=SER=BOS005,DISP=(NEW,KEEP), 00003700
// DCB=(RECFM=FB,BLKSIZE=4000,LRECL=100); 00003800
// SPACE=(3200,(2500,50)) 00003900
//GO.SYSPRINT DD SYSOUT=A 00004000
//GO.PR DD SYSOUT=A 00004100
// 00004200
```

A33,86.055,18:45:59,A33.PROGRAM.LIBRARY(BUMERGE)

```
//A33VHUM2 JOB TYPRUN=HOLD,CLASS=A,MSGCLASS=X,NOTIFY=A33          00000100
//BUMERGE EXEC PGM=SORT,PARM=' SIZE(MAX)'                        00000200
//SYSOUT DD SYSOUT=A                                             00000300
//SYSPRINT DD SYSOUT=A                                           00000400
//SORTIN DD DSN=SVHTAPE,UNIT=TAPE,DISP=(OLD,KEEP),LABEL=(2,BLP), 00000500
//          DCB=(RECFM=FB,BLKSIZE=4000,LRECL=100),VOL=SER=TN3788 00000600
//          DD DSN=A33.VH.TEM(SVHB0693),UNIT=DISK,DISP=(OLD,KEEP), 00000700
//          DCB=(RECFM=FB,BLKSIZE=4000,LRECL=100),VOL=SER=BOS005 00000800
//SORTOUT DD DSN=SVHTAAT,UNIT=TAPE,DISP=(NEW,KEEP),              00000900
//          DCB=(RECFM=FB,BLKSIZE=4000,LRECL=100),VOL=SER=SCRTCH 00001000
//SORTWK01 DD UNIT=DISK,VOL=SER=SORTWK,SPACE=(CYL,(100))         00001100
//SYSIN DD *                                                       00001200
SORT FIELDS=(10,4,A,15,8,A,14,1,A,23,4,A),FORMAT=CH             00001300
/*                                                                 00001400
//                                                                 00001500
```

A33,86.055,17:53:01,A33.PROGRAM.LIBRARY(BYUNTAB)

```
//A33VHWT8 JOB CLASS=A,MSGCLASS=X,NOTIFY=A33
//BYUNTAB EXEC PLIFCLG
//PLI.SYSIN DD *
PROCESS S,GS,NEST,OPT(2);
LEE: PROC OPTIONS(MAIN);
    DCL TPI FILE RECORD INPUT;
    DCL I TA,
        2 CD          CHAR(1),
        2 CR          CHAR(5),
        2 YR          CHAR(3),
        2 GS          CHAR(4),
        2 ER          CHAR(1),
        2 SS          CHAR(8),
        2 EE          PIC'(2)9',
        2 CARD        CHAR(76);
    DCL TAB(3)        FLOAT DEC(9) INIT(0);
    DCL TOT(3)        FLOAT DEC(9) INIT(0);
    DCL CNT          CHAR(4)      INIT('0');
    DCL SW           CHAR(1)      INIT('0');
    DCL TIT          CHAR(40)     INIT('NUMBER
JUNIP JUNCHUL');
    ON ENDFILE(TPI) GO TO LAST;
    OPEN FILE(TPI), FILE(SYSPRINT) LINESIZE(132);
    PUT PAGE;
    PUT SKIP(3) EDIT('TABLE : 1')(X(5),A(9));
    PUT SKIP(3) EDIT(TIT)(X(5),A);
    PUT SKIP(1) EDIT((40)'-')(X(5),A(40));
    TAB = 0;
RD:   READ FILE(TPI) INTO(TA);
    IF YR ^= '507' THEN GO TO RD;
    IF SW ^= '0' THEN GO TO NS;
    SW = '1';
    CNT = GS;
NS:   IF CNT ^= GS THEN GO TO WRITE;
RET:  IF ER ^= '1' THEN GO TO G1;
    TAB(2) = TAB(2) + EE;
    TAB(1) = TAB(1) + EE;
    GO TO RD;
G1:   IF ER ^= '2' THEN GO TO G2;
    TAB(3) = TAB(3) + EE;
    TAB(1) = TAB(1) + EE;
    GO TO RD;
G2:   IF ER ^= '3' THEN GO TO G3;
    TAB(2) = TAB(2) + 1;
    TAB(1) = TAB(1) + 1;
    GO TO RD;
G3:   IF ER ^= '4' THEN GO TO RD;
    TAB(3) = TAB(3) + 1;
    TAB(1) = TAB(1) + 1;
    GO TO RD;
LAST: LST = '1';
WRITE:
    PUT SKIP(2) EDIT(CNT,(TAB(N) DO N = 1 TO 3))
        (X(8),A,X(4),3 F(10));
    DO K = 1 TO 6;
        TOT(K) = TOT(K) + TAB(K);
    END;
    TAB = 0;
    IF LST = '1' THEN GO TO CLOSE;
    CNT = GS;
    GO TO RET;
CLOSE:
    PUT SKIP(3) EDIT('TOTAL',(TOT(L) DO L=1 TO 3))
TOTAL
```

```
                (X(6),A(5),X(4),3 F(10));
CLOSE FILE(TPI);
END LEE;
/*
//GO.TPI DD UNIT=DISK,DSN=A33.VH.TEM(SVHB0793),VOL=SER=80S005,
// DCB=(RECFM=FB,LRECL=100,BLKSIZE=4000),DISP=SHR
//GO.SYSPRINT DD SYSOUT=*
//
```

```
00006400
00006500
00006600
00006700
00006800
00006900
00007000
00007100
```



A33,86.055,17:26:47,A33.PROGRAM.LIBRARY(JOSORT)

```
//H01VHUSC JOB CLASS=A,MSGCLASS=X,NOTIFY=H01,TYPRUN=HOLD 00000100
//SORT EXEC PGM=SORT,PARM='SIZE(MAX)' 00000200
//SYSOUT DD SYSOUT=* 00000300
//SYSPRINT DD SYSOUT=* 00000400
//SORTIN DD DSN=SVHTAPE,UNIT=TAPE,DISP=(OLD,KEEP),LABEL=(,NL), 00000500
// DCB=(RECFM=F,LRECL=100,BLKSIZE=100,EROPT=SKP), 00000600
// VOL=SER=TN3481 00000700
//SORTOUT DD DSN=A33.VH.TEM(SVHJ0993),UNIT=DISK,DISP=(OLD,KEEP), 00000800
// VOL=SER=BOS005 00000900
//** DCB=(RECFM=FB,LRECL=80,BLKSIZE=3200),SPACE=(3200,(5,5)) 00001000
//SORTWK01 DD UNIT=DISK,SPACE=(CYL,(100)),VOL=SER=SORTWK 00001100
//SYSIN DD * 00001200
SORT FIELDS=(7,12,A,19,1,A),FORMAT=BI,FILSZ=E60000 00001300
/* 00001400
// 00001500
```

A33,86.055,17:27:07,A33.PROGRAM.LIBRARY(JOEDIT)

```
//A33VHWP JOB CLASS=H,MSGCLASS=X,NOTIFY=A33 00000100
//JSCHECK EXEC PLIFCLG 00000200
//PLI.SYSIN DD * 00000300
* PROCESS S,GS,NEST,OPT(2); 00000400
RIVER: PROC OPTIONS(MAIN); 00000500
DCL TPI FILE RECORD INPUT ; 00000600
DCL 1 TA , 00000700
    2 CARD CHAR(1), 00000800
    2 GI CHAR(17), 00000900
    2 GUBUN CHAR(1), 00001000
    2 GITA CHAR(81); 00001100
DCL 1 CA1 DEF GITA, 00001200
    2 MNAME CHAR(20), 00001300
    2 MNO CHAR(2), 00001400
    2 CNAME CHAR(20), 00001500
    2 CHY CHAR(2), 00001600
    2 CHM CHAR(2), 00001700
    2 CHD CHAR(2), 00001800
    2 CSEX CHAR(1), 00001900
    2 MAGE CHAR(2), 00002000
    2 CHSEQ CHAR(2), 00002100
    2 IMY CHAR(2), 00002200
    2 MED CHAR(2), 00002300
    2 JOSAN CHAR(1), 00002400
    2 BLK1 CHAR(22); 00002500
DCL 1 CA2 DEF GITA, 00002600
    2 SNAME CHAR(20), 00002700
    2 GAWON CHAR(2), 00002800
    2 SSEX CHAR(1), 00002900
    2 BIY CHAR(3), 00003000
    2 BIM CHAR(2), 00003100
    2 BID CHAR(2), 00003200
    2 SABY CHAR(3), 00003300
    2 SABIM CHAR(2), 00003400
    2 SABID CHAR(2), 00003500
    2 SAED CHAR(3), 00003600
    2 JKU CHAR(2), 00003700
    2 HONS CHAR(1), 00003800
    2 JIN CHAR(1), 00003900
    2 SAWO CHAR(2), 00004000
    2 BLK2 CHAR(35); 00004100
DCL 1 CA3 DEF GITA , 00004200
    2 BNAME CHAR(20), 00004300
    2 BNO CHAR(2), 00004400
    2 BAGE CHAR(2), 00004500
    2 BED CHAR(3), 00004600
    2 CHNAME CHAR(20), 00004700
    2 CHGNO CHAR(2), 00004800
    2 CHAGE CHAR(2), 00004900
    2 CHED CHAR(3), 00005000
    2 HOY CHAR(2), 00005100
    2 HOM CHAR(2), 00005200
    2 BLK3 CHAR(23); 00005300
DCL 1 CA4 DEF GITA , 00005400
    2 PNAME CHAR(20), 00005500
    2 BUAGE CHAR(2), 00005600
    2 CHNAME CHAR(20), 00005700
    2 CHEAGE CHAR(2), 00005800
    2 JASU CHAR(2), 00005900
    2 HONY CHAR(2), 00006000
    2 HONM CHAR(2), 00006100
    2 Baly CHAR(2), 00006200
    2 BALM CHAR(2), 00006300
```

```

          2 BLK4          CHAR(27);          00006400
DCL BNT          PIC(4)9' INIT(0);          00006500
DCL SWE          CHAR(1) INIT('0');        00006600
DCL ANT          PIC(4)9' INIT(0);          00006700
DCL SAWW(12)     CHAR(2) INIT('11','21','29','31',
          '39','41','49','51','61','71','81','99'); 00006800
DCL TTT          CHAR(100) INIT('.....5...10...15...20...
25...30...35...40...45...50...55...60...65...70...75...80...85...90...
5...100');          00006900
DCL ERR          CHAR(90) INIT('');          00007000
OPEN FILE(TPI), FILE(SYSPRINT) LINESIZE(132); 00007100
ON ENDFILE(TPI) GO TO LAST;                00007200
PUT SKIP(3) EDIT(TTT)(X(14),A);            00007300
RD: READ FILE(TPI) INTO (TA);               00007400
ANT = ANT + 1;                              00007500
IF CARD = '3' THEN GO TO AA;                00007600
SUBSTR(ERR,1,1)='*';                         00007700
AA: IF GUBUN = '1' THEN GO TO CARD1;         00007800
IF GUBUN = '2' THEN GO TO CARD2;           00007900
IF GUBUN = '3' THEN GO TO CARD3;           00008000
IF GUBUN = '4' THEN GO TO CARD4;           00008100
PUT SKIP(2) EDIT('NOTHING CARD',CARD,GI,GUBUN,GITA)
          (X(20),A(12),X(2),4-A);          00008200
GO TO RD;                                    00008300
CARD1:                                        00008400
IF CHY < '84' |                                00008500
  CHM > '85' & CHM < '99' THEN DO;            00008600
  SWE = '1';                                  00008700
  SUBSTR(ERR,62,2) = '**';                    00008800
  END;                                         00008900
IF CHM < '01' |                                00009000
  CHM > '12' & CHM < '99' THEN DO;            00009100
  SWE = '1';                                  00009200
  SUBSTR(ERR,64,2) = '**';                    00009300
  END;                                         00009400
IF CHD < '01' |                                00009500
  CHD > '31' & CHD < '99' THEN DO;            00009600
  SWE = '1';                                  00009700
  SUBSTR(ERR,66,2) = '**';                    00009800
  END;                                         00009900
IF CSEX < '1' |                                00010000
  CSEX > '2' & CSEX < '9' THEN DO;            00010100
  SWE = '1';                                  00010200
  SUBSTR(ERR,68,1) = '*';                     00010300
  END;                                         00010400
IF MAGE < '15' |                                00010500
  MAGE > '55' & MAGE < '99' THEN DO;         00010600
  SWE = '1';                                  00010700
  SUBSTR(ERR,69,2) = '**';                    00010800
  END;                                         00010900
IF CHSEQ < '01' |                                00011000
  CHSEQ > '10' & CHSEQ < '99' THEN DO;       00011100
  SWE = '1';                                  00011200
  SUBSTR(ERR,71,2) = '**';                    00011300
  END;                                         00011400
IF IMY < '28' |                                00011500
  IMY > '43' & IMY < '99' THEN DO;           00011600
  SWE = '1';                                  00011700
  SUBSTR(ERR,73,2) = '**';                    00011800
  END;                                         00011900
IF MED < '000' |                                00012000
  MED > '183' & MED < '999' THEN DO;         00012100
  SWE = '1';                                  00012200
  END;                                         00012300

```

```

SUBSTR(ERR,75,3) = '***'; 00012700
END; 00012800
IF JOSAN < '1' | 00012900
  JOSAN > '3' & JOSAN < '9' THEN DO; 00013000
  SWE = '1'; 00013100
  SUBSTR(ERR,78,1) = '*'; 00013200
END; 00013300
GO TO BACK; 00013400
CARD2: 00013500
IF GAWON = '00' THEN DO; 00013600
  SUBSTR(ERR,40,2) = '**'; 00013700
  SWE = '1'; 00013800
END; 00013900
IF SSEX < '1' | 00014000
  SSEX > '2' & SSEX < '9' THEN DO; 00014100
  SWE = '1'; 00014200
  SUBSTR(ERR,42,1) = '*'; 00014300
END; 00014400
IF BIY < '870' | 00014500
  BIY > '985' & BIY < '999' THEN DO; 00014600
  SWE = '1'; 00014700
  SUBSTR(ERR,43,3) = '***'; 00014800
END; 00014900
IF BIM < '01' | 00015000
  BIM > '12' & BIM < '99' THEN DO; 00015100
  SWE = '1'; 00015200
  SUBSTR(ERR,46,2) = '**'; 00015300
END; 00015400
IF BID < '01' | 00015500
  BID > '31' & BID < '99' THEN DO; 00015600
  SWE = '1'; 00015700
  SUBSTR(ERR,48,2) = '**'; 00015800
END; 00015900
IF SABIY < '984' | 00016000
  SABIY > '985' & BIY < '999' THEN DO; 00016100
  SWE = '1'; 00016200
  SUBSTR(ERR,50,3) = '***'; 00016300
END; 00016400
IF SABIM < '01' | 00016500
  SABIM > '12' & SABIM < '99' THEN DO; 00016600
  SWE = '1'; 00016700
  SUBSTR(ERR,53,2) = '**'; 00016800
END; 00016900
IF SABID < '01' | 00017000
  SABID > '31' & SABID < '99' THEN DO; 00017100
  SWE = '1'; 00017200
  SUBSTR(ERR,55,2) = '**'; 00017300
END; 00017400
IF SAED < '000' | 00017500
  SAED > '183' & SAED < '999' THEN DO; 00017600
  SWE = '1'; 00017700
  SUBSTR(ERR,57,3) = '***'; 00017800
END; 00017900
IF JKU < '00' | 00018000
  JKU > '07' & JKU < '99' THEN DO; 00018100
  SWE = '1'; 00018200
  SUBSTR(ERR,60,2) = '**'; 00018300
END; 00018400
IF HONS < '1' | 00018500
  HONS > '4' & HONS < '9' THEN DO; 00018600
  SWE = '1'; 00018700
  SUBSTR(ERR,62,1) = '*'; 00018800
END; 00018900

```

```

IF JIN < '1' |
  JIN > '3' & JIN < '9' THEN DO;
  SWE = '1';
  SUBSTR(ERR,63,1) = '*';
  END;
DO N = 1 TO 12;
  IF SAWO = _SAWW(N) THEN GO TO BACK;
  END;
SWE = '1';
SUBSTR(ERR,64,2) = '**';
GO TO BACK;
CARD3:
IF BAGE < '15' | BAGE > '50' THEN DO;
  SWE = '1';
  SUBSTR(ERR,42,2) = '**';
  END;
IF BED < '000' |
  BED > '183' & BED < '999' THEN DO;
  SWE = '1';
  SUBSTR(ERR,44,3) = '***';
  END;
IF CHAGE < '15' | CHAGE > '50' THEN DO;
  SWE = '1';
  SUBSTR(ERR,69,2) = '**';
  END;
IF CHED < '000' |
  CHED > '183' & CHED < '999' THEN DO;
  SWE = '1';
  SUBSTR(ERR,71,3) = '***';
  END;
IF HOY < '84' |
  HOY > '85' & HOY < '99' THEN DO;
  SWE = '1';
  SUBSTR(ERR,74,2) = '**';
  END;
IF HOM < '01' |
  HOM > '12' & HOM < '99' THEN DO;
  SWE = '1';
  SUBSTR(ERR,76,2) = '**';
  END;
GO TO BACK;
CARD4:
IF BUAGE < '15' | BUAGE > '98' THEN DO;
  SWE = '1';
  SUBSTR(ERR,40,2) = '**';
  END;
IF CHEAGE < '15' |
  CHEAGE > '98' THEN DO;
  SWE = '1';
  SUBSTR(ERR,42,2) = '**';
  END;
IF JASU > '10' THEN DO;
  SWE = '1';
  SUBSTR(ERR,64,2) = '**';
  END;
IF HONY < '20' |
  HONY > '85' & HONY < '99' THEN DO;
  SWE = '1';
  SUBSTR(ERR,66,2) = '**';
  END;
IF HONM < '01' |
  HONM > '12' & HONM < '99' THEN DO;
  SWE = '1';

```

```

00019000
00019100
00019200
00019300
00019400
00019500
00019600
00019700
00019800
00019900
00020000
00020100
00020200
00020300
00020400
00020500
00020600
00020700
00020800
00020900
00021000
00021100
00021200
00021300
00021400
00021500
00021600
00021700
00021800
00021900
00022000
00022100
00022200
00022300
00022400
00022500
00022600
00022700
00022800
00022900
00023000
00023100
00023200
00023300
00023400
00023500
00023600
00023700
00023800
00023900
00024000
00024100
00024200
00024300
00024400
00024500
00024600
00024700
00024800
00024900
00025000
00025100
00025200

```

```

SUBSTR(ERR,68,2) = '**';00025300
END;00025400
IF BALY 7= '85' THEN DO;00025500
    SWE = '1';00025600
    SUBSTR(ERR,70,2) = '**';00025700
END;00025800
IF BALM < '01' |00025900
    BALM > '12' & BALM < '99' THEN DO;00026000
    SWE = '1';00026100
    SUBSTR(ERR,72,2) = '**';00026200
END;00026300
BACK: IF SWE = '0' THEN GO TO RD;00026400
IF GUBUN 7='1' THEN GO TO PR1;00026500
PUT SKIP(2) EDIT(ERR)(X(14),A);00026600
PUT SKIP(2) EDIT('1 CARD',CARD,GI,GUBUN,SUBSTR(GITA,21,2),00026700
    SUBSTR(GITA,43,17))00026800
    (X(6),A(6),X(2),A,A,A,X(20),A,X(20),A);00026900
CALL HPUT2('OS',0,33,20,SUBSTR(GITA,1,20),00027000
    2,20,SUBSTR(GITA,23,20));00027100
GO TO CLEAR;00027200
PR1: IF GUBUN 7='2' THEN GO TO PR2;00027300
PUT SKIP(2) EDIT(ERR)(X(14),A);00027400
PUT SKIP(2) EDIT('2 CARD',CARD,GI,GUBUN,SUBSTR(GITA,21,26))00027500
    (X(6),A(6),X(2),A,A,A,X(20),A);00027600
CALL HPUT1('OS',0,33,20,SUBSTR(GITA,1,20));00027700
GO TO CLEAR;00027800
PR2: IF GUBUN 7= '3' THEN GO TO PR3;00027900
PUT SKIP(2) EDIT(ERR)(X(14),A);00028000
PUT SKIP(2) EDIT('3 CARD',CARD,GI,GUBUN,SUBSTR(GITA,21,7),00028100
    SUBSTR(GITA,49,11))00028200
    (X(6),A(6),X(2),A,A,A,X(20),A,X(20),A);00028300
CALL HPUT2('OS',0,33,20,SUBSTR(GITA,1,20),00028400
    7,20,SUBSTR(GITA,28,20));00028500
GO TO CLEAR;00028600
PR3: PUT SKIP(2) EDIT (ERR) (X(14),A);00028700
PUT SKIP(2) EDIT('4 CARD',CARD,GI,GUBUN,SUBSTR(GITA,21,2),00028800
    SUBSTR(GITA,43,17))00028900
    (X(6),A(6),X(2),A,A,A,X(20),A,X(20),A);00029000
CALL HPUT2('OS',0,33,20,SUBSTR(GITA,1,20),00029100
    2,20,SUBSTR(GITA,23,20));00029200
CLEAR:00029300
ERR = ' ';00029400
BNT = BNT + 1;00029500
SWE = '0';00029600
GO TO RD;00029700
LAST: PUT SKIP(3) EDIT('TOTAL = ',ANT,'ERROR = ',BNT)00029800
    (X(30),A(8),A,X(5),A(8),A);00029900
CLOSE FILE(TPI);00030000
% INCLUDE HANPRTS ;00030100
END RIVER;00030200
/*00030300
//GO.TPI DD DSN=A33.VH.TEM(SVHJ0893),UNIT=DISK,VOL=SER=BOS005,00030400
// DCB=(RECFM=FB,LRECL=100,BLKSIZE=4000),DISP=OLD00030500
//GO.SYSPRINT DD SYSOUT=*00030600
//00030700

```

A33,86.055,18:06:13,A33.PROGRAM.LIBRARY(SVHWCP33)

```
//A33VHWCP JOB TYPRUN=HOLD,MSGCLASS=X,NOTIFY=A33 00000102
//STEP1 EXEC PLIFCLG 00000200
//PLI.SYSIN DD * 00000300
* PROCESS S,GS,A,XREF,OPT(2); 00000400
LEE: PROC OPTIONS(MAIN); 00000500
    DCL MAS FILE RECORD INPUT; 00000600
    DCL NEW FILE RECORD OUTPUT; 00000700
    DCL TA CHAR(100); 00000802
    DCL I IN DEF TA, 00000900
        2 AA CHAR(6); 00001000
        2 GI CHAR(15); 00001100
        2 FF CHAR(23); 00001200
        2 BLKE CHAR(56); 00001302
    DCL MM CHAR(100); 00001402
    ON ENDFILE(MAS) GO TO CLOS; 00001500
    ON ENDFILE(SYSIN) GO TO PASS; 00001600
    OPEN FILE(MAS), FILE(NEW), FILE(SYSPRINT) LINESIZE(132); 00001700
RD: GET SKIP EDIT(MM)(A(100)); 00001802
    SUBSTR(TA,1,100) = MM; 00001902
    WRITE FILE(NEW) FROM(TA); 00002000
    GO TO RD; 00002100
PASS: READ FILE(MAS) INTO(TA); 00002200
/* ***** CHANGE *****/ 00002300
    IF GI = '8774090101104' THEN SUBSTR(FF,9,2) = '40' ; 00002400
    WRITE FILE(NEW) FROM(TA); 00002500
    GO TO PASS ; 00002600
CLOS: CLOSE FILE(MAS), FILE(NEW); 00002700
END LEE; 00002800
/* 00002900
//GO.SYSIN DD * 00003000
/* 00003100
//GO.MAS DD DSN=SVHJ152,UNIT=DISK,VOL=SER=B0S005,DISP=(OLD,KEEP), 00003202
// DCB=(RECFM=FB,LRECL=100,BLKSIZE=4000) 00003302
//GO.NEW DD DSN=SVHJ252,UNIT=DISK,VOL=SER=B0S005,DISP=(NEW,KEEP), 00003402
// DCB=(RECFM=FB,LRECL=100,BLKSIZE=4000),SPACE=(3200,(200,50)) 00003502
//GO.SYSPRINT DD SYSOUT=A 00003600
// 00003700
```

A33,86.055,18:46:13,A33.PROGRAM.LIBRARY(JOMERGE)

```
//A33VHUM3 JOB TYPRUN=HOLD,MSGCLASS=X,CLASS=A,NOTIFY=A33 00000100
//JOMERGE EXEC PGM=SORT,PARM='SIZE(MAX)' 00000200
//SYSOUT DD SYSOUT=A 00000300
//SYSPRINT DD SYSOUT=A 00000400
//SORTIN DD DSN=S93VHTP,UNIT=TAPE,DISP=(OLD,KEEP),LABEL=(2,BLP), 00000500
// DCB=(RECFM=FB,BLKSIZE=4000,LRECL=100),VOL=SER=TN4461 00000600
// DD DSN=A33.VH.TEM(SVHJ0693),UNIT=DISK,DISP=(OLD,KEEP), 00000700
// DCB=(RECFM=FB,BLKSIZE=4000,LRECL=100),VOL=SER=BOS005 00000800
//SORTOUT DD DSN=SVHBBPE,UNIT=TAPE,DISP=(NEW,KEEP), 00000900
// DCB=(RECFM=FB,BLKSIZE=4000,LRECL=100),VOL=SER=SCRTCH 00001000
//SORTWK01 DD UNIT=DISK,VOL=SER=SORTWK,SPACE=(CYL,(100)) 00001100
//SYSIN DD * 00001200
SORT FIELDS=(7,12,A,19,1,A),FORMAT=CH 00001300
/* 00001400
// 00001500
```



A33,86.055,17:53:11,A33.PROGRAM.LIBRARY(JOSATAB)

```
//A33VHTTA JOB CLASS=A,MSGCLASS=X,NOTIFY=A33
//JOSATAB EXEC COBUCLG
//COB.SYSIN DD *
IDENTIFICATION DIVISION.
PROGRAM-ID.      JOSA.
ENVIRONMENT      DIVISION.
CONFIGURATION    SECTION.
SPECIAL-NAMES.
    C01 IS PAGE-SKIP.
INPUT-OUTPUT     SECTION.
FILE-CONTROL.
    SELECT TAPE-F ASSIGN TO UT-S-INPUT.
    SELECT PRINT-F ASSIGN TO UT-S-OUTPUT.
DATA DIVISION.
FILE SECTION.
FD TAPE-F
BLOCK CONTAINS 40 RECORDS
RECORD CONTAINS 100 CHARACTERS
LABEL RECORD IS STANDARD
DATA RECORD IS TAPE-REC.
01 TAPE-REC.
    03 FIL PIC X(6).
    03 JOSA PIC 9999.
    03 FIL PIC X(8).
    03 GUBUN PIC 9.
    03 CHUL.
        05 FIL PIC X(42).
        05 CYM PIC X(4).
        05 FIL PIC X(35).
    03 SA REDEFINES CHUL.
        05 FIL PIC X(30).
        05 SYM PIC X(5).
        05 FIL PIC X(46).
    03 HON REDEFINES CHUL.
        05 FIL PIC X(54).
        05 HYM PIC X(4).
        05 FIL PIC X(23).
    03 IHON REDEFINES CHUL.
        05 FIL PIC X(50).
        05 IYM PIC X(4).
        05 FIL PIC X(27).
FD PRINT-F LABEL RECORD OMITTED.
01 PRINT-REC PIC X(132).
WORKING-STORAGE SECTION.
77 SW PIC 9 VALUE 1.
77 J PIC 999 VALUE 1.
77 I PIC 9 VALUE 1.
77 K PIC 999 VALUE 1.
77 L PIC 9 VALUE 0.
77 P PIC 9 VALUE 0.
77 LN-CNT PIC 999 VALUE 0.
77 SJOSA PIC 9999.
01 ARR-REC.
    03 FIL OCCURS 600 TIMES.
        05 ARR OCCURS 5 TIMES PIC 9(4).
01 TOT-REC1 VALUE ZEROS.
    03 TOT PIC 99999 OCCURS 4.
01 TOT-REC.
    03 FIL PIC X(15) VALUE SPACE.
    03 FIL PIC X(5) VALUE "TOTAL".
    03 TOT-T PIC B(5)Z(9)9 OCCURS 4.
01 HEAD-REC.
    03 FIL PIC X(40) VALUE SPACE.
```

03	H-R	PIC	X(26).			00006400	
03	FIL	PIC	X(66)	VALUE	SPACE.	00006500	
01	SUB-HEAD.						00006600
03	FIL	PIC	X(10)	VALUE	SPACE.	00006700	
03	FIL	PIC	X(10)	VALUE	" JOSA-NO".	00006800	
03	FIL	PIC	X(5)	VALUE	SPACE.	00006900	
03	FIL	PIC	X(10)	VALUE	" BIRTH".	00007000	
03	FIL	PIC	X(5)	VALUE	SPACE.	00007100	
03	FIL	PIC	X(10)	VALUE	" DEATH".	00007200	
03	FIL	PIC	X(5)	VALUE	SPACE.	00007300	
03	FIL	PIC	X(10)	VALUE	" MARRIAGE".	00007400	
03	FIL	PIC	X(5)	VALUE	SPACE.	00007500	
03	FIL	PIC	X(10)	VALUE	" DIVORCE".	00007600	
01	PR-REC.						00007700
03	FIL	PIC	X(10)	VALUE	SPACE.	00007800	
03	JS	PIC	Z(6)9999.			00007900	
03	TAB	PIC	B(5)Z(9)9	OCCURS	4.	00008000	
PROCEDURE						00008100	
START-RTN.						00008200	
OPEN INPUT TAPE-F OUTPUT PRINT-F.						00008300	
MOVE ZEROS TO ARR-REC.						00008400	
MOVE ZEROS TO TOT(1) TOT(2) TOT(3) TOT(4).						00008500	
MOVE ALL "*" TO H-R.						00008600	
WRITE PRINT-REC FROM HEAD-REC AFTER PAGE-SKIP.						00008700	
MOVE "*** AUGUST RESULT ***" TO H-R.						00008800	
WRITE PRINT-REC FROM HEAD-REC AFTER 1.						00008900	
MOVE ALL "*" TO H-R.						00009000	
WRITE PRINT-REC FROM HEAD-REC AFTER 1.						00009100	
WRITE PRINT-REC FROM SUB-HEAD AFTER 4.						00009200	
READ-RTN.						00009300	
READ TAPE-F AT END GO TO LAST-RTN.						00009400	
IF GUBUN = 1 THEN MOVE 2 TO I GO TO TAB-RTN.						00009500	
IF GUBUN = 2 THEN MOVE 3 TO I GO TO TAB-RTN.						00009600	
IF GUBUN = 3 THEN MOVE 4 TO I GO TO TAB-RTN.						00009700	
IF GUBUN = 4 THEN MOVE 5 TO I GO TO TAB-RTN.						00009800	
GO TO READ-RTN.						00009900	
TAB-RTN.						00010000	
IF SW = 1 THEN MOVE 0 TO SW						00010100	
MOVE JOSA TO SJOSA						00010200	
MOVE JOSA TO ARR(J 1).						00010300	
IF SJOSA = JOSA THEN GO TO JOB-RTN						00010400	
ELSE MOVE JOSA TO SJOSA						00010500	
ADD 1 TO J.						00010600	
MOVE JOSA TO ARR(J 1).						00010700	
JOB-RTN.						00010800	
ADD 1 TO ARR(J 1).						00010900	
COMPUTE P = I - 1.						00011000	
ADD 1 TO TOT(P).						00011100	
GO TO READ-RTN.						00011200	
LAST-RTN.						00011300	
PERFORM TOT-RTN VARYING I FROM 1 BY 1 UNTIL I > 4.						00011400	
WRITE PRINT-REC FROM TOT-REC AFTER 3.						00011500	
PERFORM MOVE-RTN VARYING K FROM 1 BY 1 UNTIL K > J.						00011600	
CLOSE TAPE-F PRINT-F.						00011700	
STOP RUN.						00011800	
TOT-RTN.						00011900	
MOVE TOT(I) TO TOT-T(I).						00012000	
MOVE-RTN.						00012100	
MOVE ARR(K 1) TO JS.						00012200	
PERFORM PR-RTN VARYING I FROM 2 BY 1 UNTIL I > 5.						00012300	
WRITE PRINT-REC FROM PR-REC AFTER 2.						00012400	
MOVE SPACE TO PRINT-REC.						00012500	
ADD 1 TO LN-CNT.						00012600	

IF LN-CNT > 25 THEN WRITE PRINT-REC AFTER PAGE-SKIP	00012700
WRITE PRINT-REC FROM SUB-HEAD AFTER 4	00012800
MOVE 0 TO LN-CNT.	00012900
PR-RTN.	00013000
COMPUTE L = I - 1.	00013100
MOVE ARR(K I) TO TAB(L).	00013200
/*	00013300
//GO.INPUT DD DSN=A33.VH.TEM(SVHJ0893),UNIT=DISK,DISP=(OLD,KEEP),	00013400
// DCB=(RECFM=FB,LRECL=100,BLKSIZE=4000),	00013500
// VOL=SER=BO5005	00013600
//GO.OUTPUT DD SYSOUT=*	00013700
//SYSOUT DD SYSOUT=*	00013800
//SYSPRINT DD SYSOUT=*	00013900
//SYSDBOUT DD SYSOUT=*	00014000
//	00014100

A33,86.055,17:33:54,A33.PROGRAM.LIBRARY(MATCH6)

```
//A33VHWM6 JOB CLASS=A,MSGCLASS=X,NOTIFY=A33
//JOB CAT DD DSN=CATALOG.VBOS005,DISP=SHR
//STEP1 EXEC PLIFCLG,REGION=3072K
//PLI.SYSIN DD *
* PROCESS S,GS,NEST,A,A,XREF,OPT(2);
  BABO : PROCEDURE OPTIONS(MAIN);
    DCL TPI FILE RECORD KEYED ENV(VSAM),
        TA CHAR(100),
        TPP FILE RECORD INPUT,
        TB CHAR(100),
        TC CHAR(100);
    DCL (ONCODE,SUBSTR) BUILTIN;
    DCL (CNT1,CNT2,CNT3) PIC '9999' INIT(0);
    DCL (I,J,K,L,M) PIC '9999' INIT(0);
    DCL (KEYA,KEYB) CHAR(14) INIT(' ');
    DCL SAVE1(500) CHAR(100),
        SAVE2(5000) CHAR(100);
        SAVE1 = ' '; SAVE2 = ' ';
    OPEN FILE(TPI) DIRECT UPDATE,
        FILE(TPP),
        FILE(SYSPRINT) LINESIZE(132);
    ON ENDFILE(TPP) EOF=0; EOF=1;
    M = 1;
    ON KEY(TPI) BEGIN ;
      IF ONCODE = 52 & M = 1 THEN
        DO; PUT SKIP(1) EDIT ('DUP 1 ',TB) (A(12),A);
          GOTO RD;
        END;
      IF ONCODE = 52 & M = 2 THEN
        DO; PUT SKIP(1) EDIT ('DUP 2 ',TB) (A(12),A);
          GOTO RD2;
        END;
      IF ONCODE = 52 & M = 3 THEN
        DO; PUT SKIP(1) EDIT ('DUP 3 ',SAVE2(J)) (A(12),A);
          GOTO EX3;
        END;
      IF ONCODE = 51
        THEN PUT SKIP(1) EDIT ('NOT-FOUND ',TB) (A(12),A);
        GOTO RD;
    END;
    READ FILE(TPP) INTO(TB);
    DO WHILE(EOF=1);
      SELECT (SUBSTR(TB,14,3));
        WHEN ('601') CALL P601;
        WHEN ('602') CALL P602;
        WHEN ('603') CALL P603;
        WHEN ('604') CALL P604;
        OTHERWISE;
      END;
      RD: READ FILE(TPP) INTO(TB);
    END;
    CLOSE FILE(TPI),
        FILE(TPP);

    M = 2;
    OPEN FILE(TPI) SEQUENTIAL UPDATE;
    ON ENDFILE(TPI) EOF = 0; EOF = 1;
    RD2 : READ FILE(TPI) INTO(TA);
    DO WHILE(EOF=1);
      DO I=1 TO CNT1 UNTIL(
        SUBSTR(TA,7,12) = SUBSTR(SAVE1(I),10,4)
        || SUBSTR(SAVE1(I),17,8) | I > CNT1);

```

```

END;
IF I (<= CNT1 THEN DO;
  PUT SKIP EDIT('STP2 B : ',TA)(A);
  TC = TA;
  IF SUBSTR(TA,74,1) = ' ' |
    SUBSTR(TA,74,1) = '1' |
    SUBSTR(TA,74,1) = '2' |
    SUBSTR(TA,74,1) = '3' |
    SUBSTR(TA,74,1) = 'S' |
    SUBSTR(TA,74,1) = 'X' THEN
    SUBSTR(TA,74,1) = '8';
  SUBSTR(TA,96,4) = '8' || SUBSTR(SAVE1(I),7,3);
  REWRITE FILE(TPI) FROM(TA);
  PUT SKIP EDIT('STP2 A : ',TA)(A);
  CNT2 = CNT2 + 1;
  IF SUBSTR(TC,74,1) = ' ' |
    SUBSTR(TC,74,1) = '1' |
    SUBSTR(TC,74,1) = '2' |
    SUBSTR(TC,74,1) = '3' |
    SUBSTR(TC,74,1) = 'S' |
    SUBSTR(TC,74,1) = 'X' THEN
    SUBSTR(TC,74,1) = '3';
  SUBSTR(TC,96,4) = '8' || SUBSTR(SAVE1(I),7,3);
  SUBSTR(TC,11,08) = SUBSTR(SAVE1(I),27,08);
  SAVE2(CNT2) = TC;
END;
READ FILE(TPI) INTO(TA);
END;

STEP3:
M = 3;
CLOSE FILE(TPI);
OPEN FILE(TPI) DIRECT UPDATE;
DO J=1 TO CNT2;
WRITE FILE(TPI) FROM(SAVE2(J))
  KEYFROM(SUBSTR(SAVE2(J),7,14));
PUT SKIP EDIT('STEP3 : ',SAVE2(J)) (A,A);
EX3: END;
CLOSE FILE(TPI);

/**** DATA PROCESS SECTION *****/

P601: PROC;
  CNT1 = CNT1 + 1;
  SAVE1(CNT1) = TB;
EX: END P601;
P602 : PROC;
  KEYA,TC = ' ';
  KEYA = SUBSTR(TB,10,4) || SUBSTR(TB,17,10);
  READ FILE(TPI) INTO(TA) KEY(KEYA);
  TC = TA;
  PUT SKIP EDIT('P602-1 : ',TA) (X(20),A,A);
  IF SUBSTR(TA,74,1) = ' ' |
    SUBSTR(TA,74,1) = '1' |
    SUBSTR(TA,74,1) = '2' |
    SUBSTR(TA,74,1) = '3' |
    SUBSTR(TA,74,1) = 'S' |
    SUBSTR(TA,74,1) = 'X' THEN
    SUBSTR(TA,74,1) = '8';
  SUBSTR(TA,96,4) = '8' || SUBSTR(TB,7,3);
  IF SUBSTR(TC,74,1) = ' ' |
    SUBSTR(TC,74,1) = '1' |
    SUBSTR(TC,74,1) = '2' |
    SUBSTR(TC,74,1) = '3' |
    SUBSTR(TC,74,1) = 'S' |
    SUBSTR(TC,74,1) = 'X' THEN
    SUBSTR(TC,74,1) = '3';
  SUBSTR(TC,96,4) = '8' || SUBSTR(SAVE1(I),7,3);
  SUBSTR(TC,11,08) = SUBSTR(SAVE1(I),27,08);
  SAVE2(CNT2) = TC;
END;

```

```

SUBSTR(TC,74,1) = '3' | 00012700
SUBSTR(TC,74,1) = 'S' | 00012800
SUBSTR(TC,74,1) = 'X' THEN 00012900
SUBSTR(TC,74,1) = '3'; 00013000
SUBSTR(TC,96,4) = '8' || SUBSTR(TB,7,3); 00013100
REWRITE FILE(TPI) FROM(TA) KEY(KEYA); 00013200
PUT SKIP EDIT('P602-2 ',TA) (X(20),A,A); 00013300
KEYA = SUBSTR(TB,10,4) || SUBSTR(TB,27,10); 00013400
SUBSTR(TC,41,4) = SUBSTR(TB,37,4); 00013500
SUBSTR(TC,56,5) = SUBSTR(TB,43,5); 00013600
WRITE FILE(TPI) FROM(TC) KEYFROM(KEYA); 00013700
PUT SKIP EDIT('P602-3 ',TC) (X(20),A,A); 00013800
EX: END P602; 00013900
P603 : PROC; 00014000
KEYA,TC = ' '; 00014100
KEYA = SUBSTR(TB,10,4) || SUBSTR(TB,17,10); 00014200
READ FILE(TPI) INTO(TA) KEY(KEYA); 00014300
PUT SKIP EDIT('P603-1 ',TA) (X(20),A,A); 00014400
IF SUBSTR(TA,74,1) = ' ' | 00014500
SUBSTR(TA,74,1) = '1' | 00014600
SUBSTR(TA,74,1) = '2' | 00014700
SUBSTR(TA,74,1) = '3' | 00014800
SUBSTR(TA,74,1) = 'S' | 00014900
SUBSTR(TA,74,1) = 'X' THEN 00015000
SUBSTR(TA,74,1) = '8'; 00015100
SUBSTR(TA,96,4) = '8' || SUBSTR(TB,7,3); 00015200
REWRITE FILE(TPI) FROM(TA) KEY(KEYA); 00015300
PUT SKIP EDIT('P603-2 ',TA) (X(20),A,A); 00015400
EX: END P603; 00015500
P604 : PROC; 00015600
KEYA,TC = ' '; 00015700
KEYA = SUBSTR(TB,10,4) || SUBSTR(TB,17,10); 00015800
READ FILE(TPI) INTO(TA) KEY(KEYA); 00015900
TC = TA; 00016000
PUT SKIP EDIT('P604-1 ',TA) (X(20),A,A); 00016100
IF SUBSTR(TA,74,1) = ' ' | 00016200
SUBSTR(TA,74,1) = '1' | 00016300
SUBSTR(TA,74,1) = '2' | 00016400
SUBSTR(TA,74,1) = '3' | 00016500
SUBSTR(TA,74,1) = 'S' | 00016600
SUBSTR(TA,74,1) = 'X' THEN 00016700
SUBSTR(TA,74,1) = '8'; 00016800
SUBSTR(TA,96,4) = '8' || SUBSTR(TB,7,3); 00016900
REWRITE FILE(TPI) FROM(TA) KEY(KEYA); 00017000
PUT SKIP EDIT('P604-2 ',TA) (X(20),A,A); 00017100
IF SUBSTR(TC,74,1) = ' ' | 00017200
SUBSTR(TC,74,1) = '1' | 00017300
SUBSTR(TC,74,1) = '2' | 00017400
SUBSTR(TC,74,1) = '3' | 00017500
SUBSTR(TC,74,1) = 'S' | 00017600
SUBSTR(TC,74,1) = 'X' THEN 00017700
SUBSTR(TC,74,1) = '3'; 00017800
SUBSTR(TC,96,4) = '8' || SUBSTR(TB,7,3); 00017900
SUBSTR(TC,41,4) = SUBSTR(TB,37,4); 00018000
SUBSTR(TC,56,5) = SUBSTR(TB,43,5); 00018100
KEYA = SUBSTR(TB,10,4) || SUBSTR(TB,27,10); 00018200
WRITE FILE(TPI) FROM(TC) KEYFROM(KEYA); 00018300
PUT SKIP EDIT('P604-3 ',TC) (X(20),A,A); 00018400
EX: END P604; 00018500
END BABO; 00018600
/* 00018700
//GO.TPP DD UNIT=DISK,DCB=(RECFM=FB,BLKSIZE=4000,LRECL=100), 00018800
// DSN=A33.VH.TEM(SVHB0593),VOL=SER=BQS005,DISP=OLD 00018900

```

//GO.TPI DD DSN=A33.VH.MS,DISP=SHR  
//

00019000  
00019100

A33,86.055,17:33:46,A33.PROGRAM.LIBRARY(MATCH5)

```
//A33VHWM5 JOB CLASS=C,MSGCLASS=X,NOTIFY=A33 00000100
//JOB CAT DD DSN=CATALOG.VBOS005,DISP=SHR 00000200
//STEP1 EXEC PLIFCLG,REGION=2048K 00000300
//PLI.SYSIN DD * 00000400
* PROCESS GS,A,XREF; 00000500
RIVER: PROC OPTIONS(MAIN); 00000600
DCL TPI FILE RECORD KEYED ENV(VSAM); 00000700
DCL 1 TA, 00000800
  2 GI CHAR(6), 00000900
  2 JOSAGU CHAR(4), 00001000
  2 GAGU CHAR(8), 00001100
  2 MNO CHAR(2), 00001200
  2 GITA CHAR(80); 00001300
DCL 1 T1 DEF GITA, 00001400
  2 AA CHAR(2), 00001500
  2 BB CHAR(2), 00001600
  2 SS CHAR(2), 00001700
  2 HJ CHAR(6), 00001800
  2 ADDR CHAR(40), 00001900
  2 F CHAR(1), 00002000
  2 BIGO CHAR(1), 00002100
  2 FF CHAR(26); 00002200
DCL 1 T2 DEF GITA, 00002300
  2 NAME CHAR(20), 00002400
  2 DD1 CHAR(4), 00002500
  2 DD2 CHAR(11), 00002600
  2 DD3 CHAR(5), 00002700
  2 DD4 CHAR(13), 00002800
  2 BGO CHAR(1), 00002900
  2 FFA CHAR(26); 00003000
DCL TPP FILE RECORD INPUT ; 00003100
DCL 1 TB, 00003200
  2 GIA CHAR(9), 00003300
  2 NUM CHAR(4), 00003400
  2 CARD CHAR(1), 00003500
  2 ITEM CHAR(86); 00003600
DCL 1 JUN DEF ITEM, 00003700
  2 BERG CHAR(8), 00003800
  2 NOC CHAR(2), 00003900
  2 PYO CHAR(2), 00004000
  2 SJE CHAR(7), 00004100
  2 SWE CHAR(7), 00004200
  2 BLK2 CHAR(60); 00004300
DCL DCNT PIC '9999' INIT(0) ; 00004400
DCL AAE CHAR(14) INIT(''); 00004500
DCL BBE CHAR(14) INIT(''); 00004600
DCL ONCODE BUILTIN ; 00004700
DCL TEOF BIT(1), 00004800
  YES BIT(1) INIT('1'B), 00004900
  NO BIT(1) INIT('0'B); 00005000
OPEN FILE(TPI) DIRECT UPDATE ; 00005100
OPEN FILE(TPP), FILE(SYSPRINT) LINESIZE(132); 00005200
ON ENDFILE(TPP) TEOF=YES; 00005300
ON KEY(TPI) BEGIN ; 00005400
  IF ONCODE = 52 THEN 00005500
    DO; PUT SKIP(2) EDIT ('DATA DUPLICATED KEY=',BBE) 00005600
      (A,X(2),A(14)); 00005700
      PUT SKIP(1) EDIT ('BYUN_DONG ',GIA,NUM,CARD,ITEM) 00005800
      (A,X(2),A,A,A,A); 00005900
      DCNT = DCNT + 1 ; 00006000
    END ; 00006100
  ELSE IF ONCODE = 51 THEN 00006200
    DO; PUT SKIP(1) EDIT ('KEY NOT_FOUND KEY=',BBE) 00006300
```



```

                                (A,X(2),A(14));          00006400
PUT SKIP(1) EDIT ('BYUN_DONG ',GIA,NUM,CARD,ITEM) 00006500
                                (A,X(2),A,A,A,A);          00006600
                                DCNT = DCNT + 1;          00006700
                                END; GOTO DASI ;          00006800
                                END;                      00006900
                                READ FILE(TPP)-INTO(TB); 00007000
DO WHILE(TEOF=NO);              00007100
                                BBE = (14) ' ' ;          00007200
                                IF CARD = '5' THEN CALL CARDS ; 00007300
DASI: READ FILE(TPP) INTO(TB) ; 00007400
                                END;                      00007500
                                PUT SKIP EDIT(DCNT) (F(7)) ; 00007600
                                CLOSE FILE(TPI),FILE(TPP); 00007700
                                CARDS: PROC ;             00007800
                                SUBSTR(BBE,1,4) = NUM ;   00007900
                                IF PYO = '06' | PYO = '07' | PYO = '08' | PYO = '09'
                                    | PYO = '22'
                                THEN SUBSTR(BBE,5,10) = BERC || '00' ; 00008000
                                ELSE SUBSTR(BBE,5,10) = BERC || NOC ; 00008100
                                READ FILE(TPI) INTO(TA) KEY(BBE) ; 00008200
                                IF NOC=' ' THEN          00008300
DO; IF PYO = '06' THEN SUBSTR(AA,1,1) = SWE;          00008400
    IF PYO = '07' THEN SUBSTR(AA,2,1) = SWE;          00008500
    IF PYO = '08' THEN SUBSTR(BB,1,1) = SWE;          00008600
    IF PYO = '09' THEN SUBSTR(BB,2,1) = SWE;          00008700
    IF PYO = '22' THEN SS=SWE;                        00008800
                                END ;                    00008900
                                ELSE DO;                 00009000
    IF PYO = '12' THEN DD1=SWE;                       00009100
    IF PYO = '13' THEN SUBSTR(DD2,1,1)=SWE;          00009200
    IF PYO = '14' THEN SUBSTR(DD2,2,7)=SWE;          00009300
    IF PYO = '15' THEN SUBSTR(DD2,9,3)=SWE;          00009400
    IF PYO = '16' THEN SUBSTR(DD3,1,1)=SWE;          00009500
    IF PYO = '17' THEN SUBSTR(DD3,2,4)=SWE;          00009600
    IF PYO = '18' THEN SUBSTR(DD4,1,2)=SWE;          00009700
    IF PYO = '19' THEN SUBSTR(DD4,3,4)=SWE;          00009800
    IF PYO = '20' THEN SUBSTR(DD4,8,6)=SWE;          00009900
                                END ;                    00010000
                                REWRITE FILE(TPI) FROM(TA) KEY(BBE); 00010100
                                END CARDS ;              00010200
                                END RIVER;              00010300
                                /*                        00010400
//GO.TPP DD UNIT=DISK,DCB=(RECFM=FB,BLKSIZE=4000,LRECL=100), 00010500
// DSN=A33.VH.TEM(SVHB0693),VOL=SER=R05005,DISP=OLD 00010600
//GO.TPI DD DSN=A33.VH.MS,DISP=SHR 00010700
// 00010800
// 00010900
// 00011000

```

A33,86.055,17:34:04,A33.PROGRAM.LIBRARY(GINSERT)

```
//A33VHINS JOB CLASS=A,MSGCLASS=X,NOTIFY=A33          00000100
//JOB CAT DD DSN=CATALOG.VBOS005,DISP=SHR           00000200
//INSERT EXEC PLIFCLG                               00000300
//PLI.SYSIN DD *                                    00000400
* PROCESS S,GS,NEST;                                00000500
  KANG: PROC OPTIONS(MAIN);                          00000600
    DCL TPI FILE RECORD INPUT ;                     00000700
    DCL 1 TT CHAR(100);                              00000800
    DCL 1 TA DEF TT,                                 00000900
      2 CARD CHAR(6),                                00001000
      2 JOSA PIC '(4)9',                              00001100
      2 KEYA CHAR(10),                                00001200
      2 GU CHAR(6),                                   00001300
      2 ADDR CHAR(40),                                00001400
      2 FILA CHAR(34);                                00001500
    DCL 1 TB DEF TT,                                 00001600
      2 CARD1 CHAR(6),                                00001700
      2 JOSA1 PIC '(4)9',                              00001800
      2 KEYA1 CHAR(10),                               00001900
      2 NAME CHAR(20),                                00002000
      2 NUM CHAR(33),                                 00002100
      2 BLK CHAR(27);                                 00002200
    DCL TPO FILE RECORD ENV(VSAM);                  00002300
    DCL 1 TTT CHAR(100);                             00002400
    DCL 1 TAA DEF TTT ;                              00002500
      2 CARD CHAR(6),                                00002600
      2 JOSA PIC '(4)9',                              00002700
      2 KEYA CHAR(10),                                00002800
      2 GU CHAR(6),                                   00002900
      2 HJ PIC '(6)9',                                00003000
      2 OADDR CHAR(40),                               00003100
      2 FIL1 CHAR(28);                                00003200
    DCL 1 TBB DEF TTT,                               00003300
      2 CARD1 CHAR(6),                                00003400
      2 JOSA1 PIC '(4)9',                              00003500
      2 KEYA1 CHAR(10),                               00003600
      2 ONAME CHAR(20),                               00003700
      2 NUM CHAR(33),                                 00003800
      2 BLK CHAR(27);                                 00003900
    DCL HGBATCH ENTRY OPTIONS(ASSEMBLER);          00004000
    DCL LA FIXED BIN(15) INIT(40);                 00004100
    DCL LB FIXED BIN(15) INIT(40);                 00004200
    DCL LC FIXED BIN(15) INIT(16);                 00004300
    DCL LD FIXED BIN(15) INIT(10);                 00004400
    DCL FUNC CHAR(2) INIT('IF');                  00004500
    DCL ABC CHAR(20) INIT('');                     00004600
    DCL DDD CHAR(40) INIT('');                     00004700
    DCL RECODE CHAR(2) INIT(' ');                  00004800
    DCL (OCNT,ICNT,ECNT) PIC '(6)9' INIT(0);      00004900
    DCL DJOSA(662) PIC '(4)9' INIT                00005000
    (0001,0002,0003,0005,0006,0007,0009,0010,1011,2011,0013,0014,
    0015,0017,0018,0019,0021,0022,0023,0025,0026,0027,0029,0030,
    0031,0033,0034,0035,0037,0038,0039,0041,0042,0043,0045,0046,
    0047,0049,0050,0051,0053,0054,0055,0057,0058,0059,0061,0062,
    0063,0065,
    0066,0067,0069,0070,0071,0073,0074,0075,0077,0078,
    0079,0081,0082,0083,0085,0086,0087,0089,0090,0091,
    0093,0094,0095,0097,0098,0099,0101,0102,0103,0105,
    0106,0107,0109,0110,0111,0113,0114,0115,0117,0118,
    0119,0121,0122,0123,0125,0126,0127,0129,0130,0131,
    0133,0134,0135,0137,0139,0141,0142,0143,0145,0147,
    0149,0150,0151,0153,0154,0155,0157,0158,0159,0161,
    0162,1163,2163,0165,0166,0167,0169,0170,0171,0173,
```

0174,0175,0177,0178,0179,0181,0182,0183,0185,0186,	00006400
0187,0189,0190,0191,0193,0194,0195,0197,0198,0199,	00006500
0201,0202,0203,0205,0206,0209,0210,0211,0213,0214,	00006600
0215,0217,0218,0219,0221,0222,0223,0225,0226,0227,	00006700
0229,0230,0250,0251,0253,0254,0255,0257,0258,0259,	00006800
0261,0262,0263,0265,0266,0267,0269,0270,0271,0273,	00006900
0274,0275,0277,0278,0279,0281,0282,0283,0285,0286,	00007000
0287,0289,0290,0291,0293,0294,0295,0297,0298,0299,	00007100
0301,0302,0303,0305,0306,0307,0309,0310,0311,0313,	00007200
0314,0315,0317,0318,0319,0321,0322,0323,0325,0326,	00007300
0327,0329,0330,0331,0333,0334,0335,0337,0351,0353,	00007400
0354,0355,0357,0358,0359,0361,0362,0363,0365,0366,	00007500
0367,0369,0370,0371,0373,0374,0375,0377,0378,0379,	00007600
0381,0382,0383,0385,0386,0387,0389,0390,0391,0393,	00007700
0394,0395,0397,0398,0399,0401,0403,0405,0406,0407,	00007800
0409,0410,0411,0413,0414,0415,0417,0418,0419,0421,	00007900
0422,0423,0425,0426,0427,0429,0430,0431,0433,0434,	00008000
0502,0503,0505,0506,0507,0509,0510,0511,0513,0514,	00008100
0515,0517,0518,0519,0521,0522,0523,0525,0526,0527,	00008200
0529,0530,0531,0533,0534,1535,2535,3535,4535,0537,	00008300
0538,1539,2539,0541,0542,0543,0545,0546,0549,0550,	00008400
0551,0553,0554,0555,0557,0558,0559,0561,0562,0563,	00008500
0565,0566,0571,0572,0573,1574,2574,0575,0576,0577,	00008600
0578,0579,0580,0581,0582,1583,2583,3583,0584,0589,	00008700
0590,0591,0592,0593,0594,0595,0603,0605,0606,0607,	00008800
0609,0610,0611,0613,0618,0619,0621,0622,0623,0625,	00008900
0626,0627,0629,0630,0631,0633,0634,0635,0671,0672,	00009000
0673,0674,0675,0676,0680,0681,0682,0683,0684,0685,	00009100
0686,0687,0688,0689,0690,0691,0692,0701,0702,0703,	00009200
0705,0706,0707,0709,0710,1711,2711,3711,0713,0714,	00009300
0715,0717,0718,0719,0725,0726,0727,0729,0730,0731,	00009400
0733,0734,0735,0737,0738,0739,0741,0742,0743,0745,	00009500
0746,0747,0749,0750,0751,0753,0754,0755,0771,0772,	00009600
0773,0774,0775,0776,0777,0778,0779,0783,0784,0785,	00009700
0786,0787,0788,0789,0790,0791,0792,0793,0794,0795,	00009800
0796,1803,2803,0805,0806,0807,0809,0810,0811,0813,	00009900
0814,0815,0817,0818,0819,0821,0822,0827,0829,0830,	00010000
0831,0833,0834,0835,0837,0838,0839,0841,0842,0843,	00010100
0845,0846,0847,0849,0850,0851,0853,0854,0855,0856,	00010200
0859,0861,0862,0863,0865,0866,0867,0869,0871,0872,	00010300
0873,0874,0875,0876,0877,0878,0879,0880,0881,0882,	00010400
0883,0884,0885,0886,0890,0891,0892,0893,0894,0895,	00010500
0896,0897,0898,0899,0981,0982,	00010600
0903,0905,0906,0907,0992,0993,	00010700
1302,2302,3302,0338,0547,0567,1607,2607,3607,0614)	00010800
DCL HJCHK(662) PIC '(6)9' INIT(	00010900
111735,112630,112631,111540,112717,111542,111551,112636,112123,112123,	00011000
111223,112611,112534,111639,112025,111417,111412,111513,112028,111631,	00011100
111919,111513,111616,112027,111722,111529,112324,111814,112030,111820,	00011200
112134,111512,112234,111630,111917,112319,112024,112523,112020,112528,	00011300
111116,111716,111543,111222,111919,112331,111515,111728,111228,111627,	00011400
112518,111532,111537,112020,112517,112120,111114,111616,111813,112031,	00011500
111428,111546,111536,111421,111514,111723,112034,111128,112422,112235,	00011600
111227,111437,111120,111550,111228,111728,112212,112525,111823,111629,	00011700
112119,111711,112414,111811,112319,111548,111428,112028,111522,112323,	00011800
111520,111416,111412,112517,112222,111548,112526,111930,111540,111926,	00011900
112635,112411,112725,111630,111636,112022,111620,112613,112528,112514,	00012000
111217,111927,111621,112117,111124,112530,112425,111325,111439,111627,	00012100
111635,112619,112619,112525,111739,111613,111741,112011,111515,111226,	00012200
112123,111745,111734,112114,112715,112626,112612,111415,111632,112720,	00012300
112629,112314,111121,112714,112725,112615,112714,112616,112626,112626,	00012400
112625,112616,111631,112217,111433,112320,111222,111318,111518,112416,	00012500
112323,112226,111726,112318,111631,111219,111427,112233,111815,112331,	00012600

```

112314,112718,211227,211827,211228,211828,211828,211818,211424,212021, 00012700
211511,211425,211538,211221,211630,211523,211114,212013,211619,211526, 00012800
211615,212021,211615,211727,211626,211641,211226,211537,211329,211637, 00012900
211732,211321,211644,211623,212017,211617,211516,211411,211614,211822, 00013000
211617,211651,211739,211619,211619, 00013100
211535,211629,211733,211826,211832,211725,211637,211712,211636,211922, 00013200
211223,212013,211533,211617,211819,211424,211415,211624,211526,211718, 00013300
211817,211716,211424,221628,221235,221222,221325,221225,221324,221331, 00013400
221320,221412,221215,221514,221211,221318,221217,221128,221523,221521, 00013500
221611,221214,221126,221319,221328,221523,221411,221326,221129,221217, 00013600
221215,221331,221315,221516,221417,221516,221323,221211,221334,221521, 00013700
221514,231219,231423,231432,231319,231415,231220,231115,231119,231113, 00013800
231318,231313,231313,231328,231413,231322,231417,231219,231212,231326, 00013900
231427,231318,231424,231419,231316,311712,311714,311114,311127,311120, 00014000
311115,311126,311237,311218,311239,311218,311213,311211, 00014100
311227,311220,311226,311320,311319,311313,311424,311413,311413,311414, 00014200
311412,311424,311520,311520,311520,311520,311511,311518,311513,311513, 00014300
311615,311614,311611,311620,311411,321127,321117,321111,321219,321220, 00014400
321223,321416,321420,321525,321417,321321,321322,321314,321612,313412, 00014500
313912,313712,314511,314511,313411,314536,314434,314737,313537,314633, 00014600
314137,314339,315033,315033,315033,313631,324411,323712,323311,323732, 00014700
323133,323432,321622,331128,331116,331118,331125,331218,331213,331314, 00014800
331313,340115,340222,340112,340129,340126,340214,340117,340224,340231, 00014900
340212,340231, 00015000
340245,341120,341112,333411,333931,333831,333636,334034,333138,344112, 00015100
343211,343111,343811,343536,344331,343736,344541,344433,344242,344032, 00015200
343137,343432,351141,351135,351127,351120,351117,351123,351126,351418, 00015300
351226,351226,351226,351412,351511,351217,351415,351312,351216,360227, 00015400
360115,360211,360122,260317,360112,360332,360220,360213,360119,360121, 00015500
360314,360119, 360324,361113,361137,361132,361117, 00015600
361122,361234,361316,361225,361213,361112,353411,353112,354334,353633, 00015700
354234,354035,353135,353841,354136,364011,364811,363211,364211,363731, 00015800
363938,364934,364133,364531,363134,363536,364641,364432,364042,371121, 00015900
371121,371134,371114,371123,371111,371223,371228,371211,371319,371519, 00016000
371320,371314,371423,371414,371420,381111,381119,381117,381120,381135, 00016100
381134,381116,381516,381143,381128,381142,381242,381230,381227,381213, 00016200
381240,381225,381216,381215,381818,381820,381423,381331,381325,381323, 00016300
381327,381612,381618,381617,381423,381427,374913,374811,373411,375311, 00016400
373912,374332,374734,374432,373343,374536,373731,373440,374034,374634, 00016500
375131,374032,383412,383612,383735,384431,384834,383334,384737,384242, 00016600
384336,383836,383146,384537,391119,391125,391114,391116,393111,393131, 00016700
211651,211651,211651,212018,311423,321122,331125,331125,331125,331121
) ;
DCL ONCODE BUILTIN ; 00017000
ON KEY(TPO) BEGIN ; 00017100
IF ONCODE = 52 THEN DO ; 00017200
    PUT SKIP EDIT ('DATA DUPLICATED', 'KEY=', SUBSTR(TT, 7, 14), TT) 00017300
        (A, A, A, X(2), A(100)); 00017400
        ECNT = ECNT + 1 ; 00017500
GO TO RD; END ; 00017600
END; 00017700
ON ENDFILE(TPI) GO TO LAST ; 00017800
OPEN FILE(TPI), FILE(TPO) DIRECT UPDATE; 00017900
RD: READ FILE(TPI) INTO(TT) ; 00018000
ICNT = ICNT+1 ; 00018100
TTT=''; 00018200
IF SUBSTR(TT, 19, 2)='00' THEN 00018300
DO ; 00018400
    CALL HGBATCH(FUNC, RECODE, ADDR, LA, DDD, LB) ; 00018500
    OADDR = DDD; 00018600
    TAA = TA , BY NAME ; 00018700
    DO I=1 TO 562; 00018800
        IF TA.JOSA=DJOSA(I) THEN HJ = HJCHK(I) ; 00018900

```

```

END;
END ;
ELSE DO;
CALL HGBATCH(FUNC,RECODE,NAME,LC,ABC,LD) ;
ONAME = ABC ;
TBB = TB , BY NAME ;
END;
IF RECODE '=' THEN DO ;
PUT SKIP(2) EDIT('HAN ERROR : ',RECODE,STRING(TT))
(A,X(2)) ;
ECNT = ECNT + 1 ;
GOTO RD ;
END ;
WRITE FILE(TPO) FROM(TTT) KEYFROM(SUBSTR(TTT,7,14)) ;
OCNT = OCNT + 1 ;
GOTO RD ;
LAST: CLOSE FILE(TPO),FILE(TPI) ;
PUT SKIP EDIT('INSERT GAGU CNT =',ICNT)(A,F(5));
PUT SKIP EDIT('ERROR GAGU CNT =',ECNT)(A,F(5));
PUT SKIP EDIT('OUTPUT GAGU CNT =',OCNT)(A,F(5));
END KANG ;
/*
//GO.TPI DD DSN=S93.VH.ACT,UNIT=DISK,VOL=SER=SPS803,
// DCB=(RECFM=FB,LRECL=100,BLKSIZE=4000),DISP=OLD
//GO.TPO DD DSN=A33.VH.MS,DISP=SHR
//GO.SYSPRINT DD SYSOUT=*
//

```

```

00019000
00019100
00019200
00019300
00019400
00019500
00019600
00019700
00019800
00019900
00020000
00020100
00020200
00020300
00020400
00020500
00020600
00020700
00020800
00020900
00021000
00021100
00021200
00021300
00021400
00021500
00021600

```

A33,86.055,17:34:22,A33.PROGRAM.LIBRARY(MATCHB)

```
//A33VHMB JOB CLASS=A,MSGCLASS=X,NOTIFY=A33 00000100
//JOB CAT DD DSN=CATALOG.VBOS005,DISP=SHR 00000200
//STEP1 EXEC PLIFCLG 00000300
//PLI.SYSIN DD * 00000400
* PROCESS GS,A,XREF; 00000500
BABO : PROCEDURE OPTIONS(MAIN); 00000600
  DCL TPI FILE RECORD KEYED ENV(VSAM), 00000700
    TA CHAR(100), 00000800
    TPP FILE RECORD INPUT, 00000900
    TB CHAR(100), 00001000
    TC CHAR(100); 00001100
  DCL (ONCODE,SUBSTR) BUILTIN; 00001200
  DCL (CNT1,CNT2) PIC '9999' INIT(0); 00001300
  DCL (I,J,K,L,M) PIC '9999' INIT(0); 00001400
  DCL ECNT PIC '9999' INIT(0); 00001500
  DCL (KEYA,KEYB) CHAR(14) INIT(' '); 00001600
  DCL SAVE1(5000) CHAR(100); 00001700
  DCL SAVE2(5000) CHAR(100); 00001800
  DCL SAVE1 = ' '; SAVE2 = ' '; 00001900
  OPEN FILE(TPI) DIRECT UPDATE, 00002000
    FILE(TPP), 00002100
    FILE(SYSPRINT) LINESIZE(132); 00002200
  ON ENDFILE(TPP) EOF=0; EOF=1; 00002300
  M = 1; 00002400
  ON KEY(TPI) BEGIN ; 00002500
    IF ONCODE = 52 & M = 1 THEN 00002600
      DO; PUT SKIP(1) EDIT ('DUP 1 ',TB) (A(12),A); 00002700
        GOTO RD; 00002800
      END; 00002900
    IF ONCODE = 52 & M = 2 THEN 00003000
      DO; PUT SKIP(1) EDIT ('DUP 2 ',SAVE1(I)) (A(12),A); 00003100
        GOTO RD2; 00003200
      END; 00003300
    IF ONCODE = 52 & M = 3 THEN 00003400
      DO; PUT SKIP(1) EDIT ('DUP 3 ',SAVE2(J)) (A(12),A); 00003500
        GOTO RD3; 00003600
      END; 00003700
    IF ONCODE = 51 00003800
      THEN PUT SKIP(1) EDIT ('NOT-FOUND ',TB) (A(12),A); 00003900
      GOTO RD; 00004000
    END; 00004100
  READ FILE(TPP) INTO(TB); 00004200
  DO WHILE(EOF=1); 00004300
    SELECT (SUBSTR(TB,14,1)); 00004400
      WHEN ('1') CALL CARD_1; 00004500
      WHEN ('2') CALL CARD_2; 00004600
      WHEN ('3') CALL CARD_3; 00004700
      WHEN ('4') CALL CARD_4; 00004800
      WHEN ('7') CALL CARD_7; 00004900
      OTHERWISE; 00005000
    END; 00005100
  RD: READ FILE(TPP) INTO(TB); 00005200
  END; 00005300
  CLOSE FILE(TPI), 00005400
    FILE(TPP); 00005500
  /*** DATA PROCESS SECTION *****/ 00005600
  00005700
  CARD_1: PROC; 00005800
    CNT1 = CNT1 + 1; 00005900
    SAVE1(CNT1) = TB; 00006000
  EX: END CARD_1; 00006100
  CARD_2 : PROC; 00006200
    CNT2 = CNT2 + 1; 00006300
```

SAVE2(CNT2) = TB;	00006400
EX: END CARD_2;	00006500
CARD_3 : PROC;	00006600
KEYA,TC = ' ';	00006700
KEYA = SUBSTR(TB,10,4)    SUBSTR(TB,15,10);	00006800
CALL CONVERT_RTN ;	00006900
SUBSTR(TC,41,27) = SUBSTR(TB,45,27) ;	00007000
SUBSTR(TC,74,1) = SUBSTR(TB,87,1) ;	00007100
SUBSTR(TC,96,4) = '8'    SUBSTR(TB,7,3);	00007200
WRITE FILE(TPI) FROM(TC) KEYFROM(KEYA);	00007300
/* PUT SKIP EDIT('3_CARD ',TC) (X(20),A,A); */	00007400
EX: END CARD_3;	00007500
CONVERT_RTN:PROC;	00007600
DCL HGBATCH ENTRY OPTIONS(ASSEMBLER) ;	00007700
DCL LA FIXED BIN(15) INIT(16) ;	00007800
DCL LB FIXED BIN(15) INIT(10) ;	00007900
DCL FUNC CHAR(2) INIT('IF') ;	00008000
DCL ABC CHAR(20) INIT('') ;	00008100
DCL RECODE CHAR(2) INIT(' ') ;	00008200
CALL HGBATCH(FUNC,RECODE,SUBSTR(TB,25,20),LA,ABC,LB);	00008300
SUBSTR(TC,21,20) = ABC ;	00008400
IF RECODE '=' THEN DO ;	00008500
PUT SKIP(2) EDIT('HAN ERROR : ',RECODE,	00008600
STRING(TB))(A,X(2)) ;	00008700
ECNT = ECNT + 1 ;	00008800
END CONVERT_RTN ;	00008900
CARD_4 : PROC;	00009000
KEYA,TC = ' ';	00009100
KEYA = SUBSTR(TB,10,4)    SUBSTR(TB,15,10);	00009200
READ FILE(TPI) INTO(TA) KEY(KEYA);	00009300
TC = TA;	00009400
SUBSTR(TC,74,1) = SUBSTR(TB,60,1) ;	00009500
SUBSTR(TC,96,4) = '8'    SUBSTR(TB,7,3);	00009600
REWRITE FILE(TPI) FROM(TC) KEY(KEYA);	00009700
/* PUT SKIP EDIT('CARD-4 ',TC) (X(20),A,A); */	00009800
EX: END CARD_4;	00009900
CARD_7 : PROC ;	00010000
KEYA,TC = ' ' ;	00010100
KEYA = SUBSTR(TB,10,4)    SUBSTR(TB,15,10) ;	00010200
READ FILE(TPI) INTO(TA) KEY(KEYA) ;	00010300
TC = TA ;	00010400
SUBSTR(TC,63,5) = SUBSTR(TB,45,5) ;	00010500
REWRITE FILE(TPI) FROM(TC) KEY(KEYA) ;	00010600
EX: END CARD_7 ;	00010700
/* 1 CARD PROCESS */	00010800
OPEN FILE(TPI) SEQUENTIAL UPDATE;	00010900
ON ENDFILE(TPI) EOF = 0 ; EOF = 1 ;	00011000
M = 2 ;	00011100
RD2 : READ FILE(TPI) INTO(TA);	00011200
DO WHILE(EOF=1);	00011300
DO I=1 TO CNT1	00011400
UNTIL (SUBSTR(TA,7,12) = SUBSTR(SAVE1(I),10,4)	00011500
SUBSTR(SAVE1(I),15,8)   I > CNT1) ;	00011600
END ;	00011700
IF I <= CNT1 THEN DO ;	00011800
IF SUBSTR(SAVE1(I),40,1) = '9'	00011900
THEN SUBSTR(TA,74,1) = 'X' ;	00012000
ELSE SUBSTR(TA,74,1) = SUBSTR(SAVE1(I),40,1) ;	00012100
SUBSTR(TA,96,4) = '8'    SUBSTR(SAVE1(I),7,3);	00012200
REWRITE FILE(TPI) FROM(TA) ;	00012300
PUT SKIP EDIT('1 CARD ',TA)(A,A(100)) ;	00012400
END ;	00012500
READ FILE(TPI) INTO(TA);	00012600

END;	00012700
CLOSE FILE(TPI);	00012800
/*     2 CARD PROCESS     */	00012900
OPEN FILE(TPI) SEQUENTIAL UPDATE;	00013000
ON ENDFILE(TPI) EOF = 0 ;    EOF = 1;	00013100
M = 3 ;	00013200
RD3 : READ FILE(TPI) INTO(TA);	00013300
DO WHILE(EOF=1);	00013400
DO J=1 TO CNT2 UNTIL	00013500
(SUBSTR(TA,7,12) = SUBSTR(SAVE2(J),10,4)	00013600
SUBSTR(SAVE2(J),15,8)   J > CNT2) ;	00013700
END ;	00013800
IF J <= CNT2 THEN DO ;	00013900
IF SUBSTR(SAVE2(J),40,1) = '9'	00014000
THEN SUBSTR(TA,74,1) = 'Y' ;	00014100
ELSE SUBSTR(TA,74,1) = SUBSTR(SAVE2(J),40,1) ;	00014200
SUBSTR(TA,96,4) = '8'    SUBSTR(SAVE2(J),7,3);	00014300
REWRITE FILE(TPI) FROM(TA) ;	00014400
PUT SKIP EDIT('2 CARD ',TA)(A,A(100)) ;	00014500
END ;	00014600
READ FILE(TPI) INTO(TA);	00014700
END;	00014800
CLOSE FILE(TPI);	00014900
END BABO;	00015000
/*	00015100
//GO.TPF DD UNIT=DISK,VOL=SER=BOS005,DSN=A33.VH.TEM(SVHB0693),	00015200
//DCB=(RECFM=FB,LRECL=100,BLKSIZE=4000),DISP=OLD	00015300
//GO.TPI DD DSN=A33.VH.MS,DISP=SHR	00015400
//	00015500



A33,86.055,17:37:26,A33.PROGRAM.LIBRARY(S93VHWT5)

```
//A33VHWT5 JOB CLASS=C,MSGCLASS=X,NOTIFY=A33 00000100
//JOB CAT DD DSN=CATALOG.VBOS005,DISP=SHR 00000200
//GATAB EXEC PLIFCLG 00000300
//PLI.SYSIN DD * 00000400
* PROCESS S,GS,NEST,OPT(2); 00000500
RIVER: PROC OPTIONS(MAIN); 00000600
DCL TPI FILE RECORD INPUT ENV(VSAM) ; 00000700
DCL 1 TA, 00000800
2 CARD CHAR(6), 00000900
2 JOSA CHAR(4); 00001000
2 GI CHAR(8), 00001100
2 NUM CHAR(2), 00001200
2 NAME CHAR(20), 00001300
2 GAA CHAR(4), 00001400
2 SEX CHAR(1), 00001500
2 BIRY PIC'(3)9', 00001600
2 BIRM PIC'(2)9', 00001700
2 A1 CHAR(5), 00001800
2 BAWO CHAR(1), 00001900
2 BL1 CHAR(17), 00002000
2 BB CHAR(1), 00002100
2 BL2 CHAR(26); 00002200
DCL TAB(6) FLOAT DEC(9) INIT((6)0); 00002300
DCL TOT(6) FLOAT DEC(9) INIT((6)0); 00002400
DCL CNT CHAR(4) INIT('0'); 00002500
DCL HOLD CHAR(8) INIT('0'); 00002600
DCL AGE PIC'(8)9' INIT(0); 00002700
DCL TIT CHAR(71) INIT(' NUMBER TOTAL 00002800
MALE FEMALE UNKNOWN WOMAN HOUSEHOLD'); 00002900
DCL PP PIC'(8)9' INIT('0'); 00003000
ON ENDFILE(TPI) GO TO LAST; 00003100
OPEN FILE(TPI), FILE(SYSPRINT) LINESIZE(132); 00003200
PUT PAGE ; 00003300
PUT SKIP(3) EDIT('TABLE : 1')(X(5),A(9)); 00003400
PUT SKIP(3) EDIT(TIT) (X(5),A); 00003500
PUT SKIP(1) EDIT((71)'-')(X(5),A(71)); 00003600
TAB = 0; 00003700
RD: READ FILE(TPI) INTO(TA); 00003800
IF NUM = '00' THEN GO TO RD ; 00003900
IF BB > '4' THEN GO TO RD; 00004000
IF BB = 'D' | BB = 'Y' THEN GO TO RD; 00004100
/* DO I = 23, 28, 33, 38; 00004200
IF SUBSTR(BL1,I,1) = '8' THEN GO TO RD; 00004300
END; */ 00004400
/* DO I = 19, 24, 29, 34; */ 00004500
/* IF SUBSTR(BL1,I,5) = '85017' THEN GO TO QQ; */ 00004600
/* END; */ 00004700
/* GO TO RD; */ 00004800
QQ: IF SW ^= '0' THEN GO TO NS; 00004900
SW = '1'; 00005000
CNT = JOSA; 00005100
NS: IF CNT ^= JOSA THEN GO TO WRITE; 00005200
RET: TAB(1) = TAB(1) + 1; 00005300
IF SEX ^= '1' THEN GO TO FEMALE; 00005400
TAB(2) = TAB(2) + 1; 00005500
FEMALE: 00005600
IF SEX ^= '2' THEN GO TO UNKNOW; 00005700
TAB(3) = TAB(3) + 1; 00005800
IF BAWO = '4' THEN GO TO NEXT1; 00005900
IF BAWO = '9' THEN GO TO NEXT1; 00006000
IF BIRY = 999 THEN GO TO NEXT1; 00006100
PP = BIRY * 12; 00006200
IF BIRM = 99 THEN GO TO PASS; 00006300
```

PP	= PP + BIRM;	00006400
PASS:		00006500
AGE	= (((985*12)+03 )- PP) / 12;	00006600
IF AGE	< 15 THEN GO TO NEXT1;	00006700
IF AGE	> 49 THEN GO TO NEXT1;	00006800
TAB(5)	= TAB(5) + 1;	00006900
GO TO NEXT1;		00007000
UNKNOW:		00007100
TAB(4)	= TAB(4) + 1;	00007200
NEXT1:		00007300
IF HOLD	= GI THEN GO TO RD;	00007400
TAB(6)	= TAB(6) + 1;	00007500
HOLD	= GI;	00007600
GO TO RD;		00007700
LAST:		00007800
LST	= '1' ;	00007900
WRITE:		00008000
PUT SKIP(2) EDIT(CNT,(TAB(N) DO N = 1 TO 6))		00008100
	(X(8),A,X(4),6 F(10));	00008200
DO K = 1 TO 6;		00008300
TOT(K)	= TOT(K) + TAB(K);	00008400
END;		00008500
TAB = 0;		00008600
IF LST = '1' THEN GO TO CLOSE;		00008700
CNT = JOSA;		00008800
GO TO RET;		00008900
CLOSE:		00009000
PUT SKIP(3) EDIT('TOTAL',(TOT(L) DO L=1 TO 6))		00009100
	(X(7),A(5),X(4),6 F(10));	00009200
CLOSE FILE(TPI);		00009300
END RIVER;		00009400
/*		00009500
//GO.TPI	DD DSN=A33.VH.MS,DISP=SHR	00009600
//GO.SYSPRINT	DD SYSOUT=*	00009700
//		00009800

A33,86.055,17:38:37,A33.PROGRAM.LIBRARY(JOCHECK)

```
//A33VHMNB JOB CLASS=A,MSGCLASS=X,NOTIFY=A33 00000100
//JOB CAT DD DSN=CATALOG.VBOS005,DISP=SHR 00000200
//STEP1 EXEC PLIFCLG 00000300
//PLI.SYSIN DD * 00000400
* PROCESS S,GS,NEST,OPT(2); 00000500
  RHEE: PROC OPTIONS(MAIN); 00000600
    DCL TPI FILE RECORD INPUT; 00000700
    DCL I TA,~ 00000800
      2 F1 CHAR(6), 00000900
      2 JOSA CHAR(4), 00001000
      2 GAGU CHAR(8), 00001100
      2 SQ CHAR(2), 00001200
      2 F3 CHAR(80); 00001300
    DCL PR CHAR(132) INIT((132)' '); 00001400
    DCL I LT DEF PR, 00001500
      2 F11 CHAR(1), 00001600
      2 OJOSA CHAR(4), 00001700
      2 F12 CHAR(2), 00001800
      2 OGAGU CHAR(8), 00001900
      2 OQ(29), 00002000
        3 F13 CHAR(2), 00002100
        3 OSQ CHAR(2), 00002200
      2 F14 CHAR(1); 00002300
    DCL SW CHAR(1) INIT('1'); 00002400
    DCL (I,J,K) PIC'99999' INIT(0); 00002500
    DCL (CNT1,CNT2) PIC'9999' INIT(0); 00002600
    DCL SGAGU CHAR(8) INIT(' '); 00002700
  DCL 00002800
  DCL 00002900
  DCL 00003000
  DCL 00003100
  DCL 00003200
  DCL 00003300
  DCL 00003400
  DCL 00003500
  DCL 00003600
  DCL 00003700
  DCL 00003800
  DCL 00003900
  DCL 00004000
  DCL 00004100
  DCL 00004200
  DCL 00004300
  DCL 00004400
  DCL 00004500
  DCL 00004600
  DCL 00004700
  DCL 00004800
  DCL 00004900
  DCL 00005000
  DCL 00005100
  DCL 00005200
  DCL 00005300
  DCL 00005400
  DCL 00005500
  DCL 00005600
  DCL 00005700
  DCL 00005800
  DCL 00005900
  DCL 00006000
  DCL 00006100
  DCL 00006200
  DCL 00006300
```

```

IF SW = '1' THEN DO;                                00006400
    SW = '0';                                       00006500
    SGAGU = GAGU;                                   00006600
    CNT1 = CNT1 + 1;                                00006700
    END;                                             00006800
IF SGAGU ^= GAGU THEN DO;                           00006900
    OJOSA = ARY(I) ;                                00007000
    OGAGU = SGAGU;                                  00007100
    PUT SKIP EDIT(PR)(A);                            00007200
    PR = (132) ' ';                                  00007300
    SGAGU = GAGU;                                    00007400
    J = J + 1;                                       00007500
    CNT1 = CNT1 + 1;                                  00007600
    END;                                             00007700
OSQ(J) = SQ;                                        00007800
J = J + 1;                                          00007900
IF SQ ^= '00' THEN CNT2 = CNT2 + 1;                00008000
GO TO RD;                                           00008100
LAST:                                               00008200
IF PR ^= (132) ' ' THEN DO;                         00008300
    OJOSA = ARY(I) ;                                00008400
    OGAGU = SGAGU;                                  00008500
    PUT SKIP EDIT(PR)(A);                            00008600
    PR = (132) ' ';                                  00008700
    PUT SKIP(3) EDIT('GAGU CNT = ',CNT1,'GAGUWON CNT = ',CNT2)00008800
    (X(10),A,F(4),X(5),A,F(4));                    00008900
    PUT SKIP(5);                                     00009000
    END;                                             00009100
    CLOSE FILE(TPI);                                00009200
END RHEE;                                           00009300
/*                                                  00009400
//GO.TPI DD UNIT=DISK,DSN=A33.VH.MS,DISP=SHR      00009500
//GO.SYSPRINT DD SYSOUT=*                          00009600
/*                                                  00009700
//                                                  00009800

```

```

A33,86.055,17:43:23,A33.PROGRAM.LIBRARY(BACKUP)

//A33VHWVV JOB CLASS=A, TYPRUN=HOLD, MSGCLASS=X, NOTIFY=A33          00000100
//JOB CAT DD DSN=CATALOG.VBOS005, DISP=SHR                          00000200
//STEP1 EXEC PLIFCLG                                                00000300
//PLI.SYSIN DD *                                                    00000400
* PROCESS GS, A, XREF, OPT(2);                                       00000500
  RIVER: PROC OPTIONS(MAIN);                                         00000600
    DCL DISKIN FILE RECORD INPUT ENV(VSAM);                          00000700
    DCL TAPEOUT FILE RECORD OUTPUT;                                  00000800
    DCL INREC          CHAR(100);                                    00000900
    DCL (ICN,WCN)      PIC'(6)9'  INIT(0);                           00001000
    ON ENDFILE(DISKIN) GO TO CLOS;                                    00001100
    OPEN FILE(DISKIN), FILE(TAPEOUT),                                00001200
        FILE(SYSPRINT) LINESIZE(132);                                00001300
  BEGIN: READ FILE(DISKIN) INTO(INREC);                               00001400
    ICN = ICN + 1;                                                  00001500
    WRITE FILE(TAPEOUT) FROM(INREC);                                  00001600
    WCN = WCN + 1;                                                  00001700
    GO TO BEGIN;                                                    00001800
  CLOS: CLOSE FILE(DISKIN), FILE(TAPEOUT);                           00001900
    PUT SKIP(5) EDIT('  INPUT  CNT = ', ICN)(A,F(6));               00002000
    PUT SKIP(5) EDIT('  BACK-UP CNT = ', WCN)(A,F(6));              00002100
  END RIVER;                                                         00002200
/*                                                                      00002300
//GO.DISKIN DD DSN=A33.VH.MS, DISP=SHR                               00002400
//GO.TAPEOUT DD UNIT=TAPE, DCB=(RECFM=FB, BLKSIZE=20000, LRECL=100), 00002500
//      DSN=SVHTT, DISP=(OLD,KEEP), VOL=SER=SCRTCH                 00002600
//GO.SYSPRINT DD SYSOUT=*                                           00002700
//GO.PR DD SYSOUT=*                                                 00002800
//                                                                      00002900

```

A33,86.055,17:44:53,A33.PROGRAM.LIBRARY(SINSERT)

```
//A33VHWM1 JOB CLASS=A,MSGCLASS=X,NOTIFY=A33
//JOB CAT DD DSN=CATALOG.VBOS005,DISP=SHR
//INSERT EXEC PLIFCLG
//PLI.SYSIN DD *
* PROCESS S,GS,NEST;
  KANG: PROC OPTIONS(MAIN);
    DCL TPI FILE RECORD INPUT ;
    DCL 1 TT CHAR(100);
    DCL 1 TA DEF TT,
      2 CARD CHAR(6),
      2 JOSA PIC '(4)9',
      2 KEYA CHAR(10),
      2 GU CHAR(6),
      2 ADDR CHAR(40),
      2 FILA CHAR(7),
      2 BIGA CHAR(1),
      2 FILB CHAR(26);
    DCL 1 TB DEF TT,
      2 CARD1 CHAR(6),
      2 JOSA1 PIC '(4)9',
      2 KEYA1 CHAR(10),
      2 NAME CHAR(20),
      2 NUM CHAR(33),
      2 BLKB CHAR(27);
    DCL TPO FILE RECORD ENV(VSAM) ;
    DCL 1 TTT CHAR(100) ;
    DCL 1 TAA DEF TTT ,
      2 CARD CHAR(6),
      2 JOSA PIC '(4)9',
      2 KEYA CHAR(10),
      2 GU CHAR(6),
      2 HJ PIC '(6)9',
      2 QADDR CHAR(40),
      2 FIL CHAR(1),
      2 BIGO CHAR(1),
      2 FILB CHAR(26);
    DCL 1 TBB DEF TTT,
      2 CARD1 CHAR(6),
      2 JOSA1 PIC '(4)9',
      2 KEYA1 CHAR(10),
      2 ONAME CHAR(20),
      2 NUM CHAR(33),
      2 BIGOA CHAR(1),
      2 BLKB CHAR(26);
    DCL HGBATCH ENTRY OPTIONS(ASSEMBLER) ;
    DCL LA FIXED BIN(15) INIT(40);
    DCL LB FIXED BIN(15) INIT(40);
    DCL LC FIXED BIN(15) INIT(16);
    DCL LD FIXED BIN(15) INIT(10);
    DCL FUNC CHAR(2) INIT('IF') ;
    DCL ABC CHAR(20) INIT('') ;
    DCL DDD CHAR(40) INIT('') ;
    DCL RECODE CHAR(2) INIT(' ') ;
    DCL (OCNT,ICNT,ECNT) PIC '(6)9' INIT(0) ;
    DCL DJOSA(572) PIC '(4)9' INIT
      (0001,0002,0003,0005,0006,0007,0009,0010,1011,2011,0013,0014,
      0015,0017,0018,0019,0021,0022,0023,0025,0026,0027,0029,0030,
      0031,0033,0034,0035,0037,0038,0039,0041,0042,0043,0045,0046,
      0047,0049,0050,0051,0053,0054,0055,0057,0058,0059,0061,0062,
      0063,0065,
      0066,0067,0069,0070,0071,0073,0074,0075,0077,0078,
      0079,0081,0082,0083,0085,0086,0087,0089,0090,0091,
      0093,0094,0095,0097,0098,0099,0101,0102,0103,0105,
      00000100
      00000200
      00000300
      00000400
      00000500
      00000600
      00000700
      00000800
      00000900
      00001000
      00001100
      00001200
      00001300
      00001400
      00001500
      00001600
      00001700
      00001800
      00001900
      00002000
      00002100
      00002200
      00002300
      00002400
      00002500
      00002600
      00002700
      00002800
      00002900
      00003000
      00003100
      00003200
      00003300
      00003400
      00003500
      00003600
      00003700
      00003800
      00003900
      00004000
      00004100
      00004200
      00004300
      00004400
      00004500
      00004600
      00004700
      00004800
      00004900
      00005000
      00005100
      00005200
      00005300
      00005400
      00005500
      00005600
      00005700
      00005800
      00005900
      00006000
      00006100
      00006200
      00006300
```

```

0106,0107,0109,0110,0111,0113,0114,0115,0117,0118,      00006400
0119,0121,0122,0123,0125,0126,0127,0129,0130,0131,      00006500
0133,0134,0135,0137,0139,0141,0142,0143,0145,0147,      00006600
0149,0150,0151,0153,0154,0155,0157,0158,0159,0161,      00006700
0162,1163,2163,0165,0166,0167,0169,0170,0171,0173,      00006800
0174,0175,0177,0178,0179,0181,0182,0183,0185,0186,      00006900
0187,0189,0190,0191,0193,0194,0195,0197,0198,0199,      00007000
0201,0202,0203,0205,0206,0209,0210,0211,0213,0214,      00007100
0215,0217,0218,0219,0221,0222,0223,0225,0226,0227,      00007200
0229,0230,0250,0251,0253,0254,0255,0257,0258,0259,      00007300
0261,0262,0263,0265,0266,0267,0269,0270,0271,0273,      00007400
0274,0275,0277,0278,0279,0281,0282,0283,0285,0286,      00007500
0287,0289,0290,0291,0293,0294,0295,0297,0298,0299,      00007600
0301,0302,0303,0305,0306,0307,0309,0310,0311,0313,      00007700
0314,0315,0317,0318,0319,0321,0322,0323,0325,0326,      00007800
0327,0329,0330,0331,0333,0334,0335,0337,0351,0353,      00007900
0354,0355,0357,0358,0359,0361,0362,0363,0365,0366,      00008000
0367,0369,0370,0371,0373,0374,0375,0377,0378,0379,      00008100
0381,0382,0383,0385,0386,0387,0389,0390,0391,0393,      00008200
0394,0395,0397,0398,0399,0401,0403,0405,0406,0407,      00008300
0409,0410,0411,0413,0414,0415,0417,0418,0419,0421,      00008400
0422,0423,0425,0426,0427,0429,0430,0431,0433,0434,      00008500
0502,0503,0505,0506,0507,0509,0510,0511,0513,0514,      00008600
0515,0517,0518,0519,0521,0522,0523,0525,0526,0527,      00008700
0529,0530,0531,0533,0534,1535,2535,3535,4535,0537,      00008800
0538,1539,2539,0541,0542,0543,0545,0546,0549,0550,      00008900
0551,0553,0554,0555,0557,0558,0559,0561,0562,0563,      00009000
0565,0566,0571,0572,0573,1574,2574,0575,0576,0577,      00009100
0578,0579,0580,0581,0582,1583,2583,3583,0584,0589,      00009200
0590,0591,0592,0593,0594,0595,0603,0605,0606,0607,      00009300
0609,0610,0611,0613,0618,0619,0621,0622,0623,0625,      00009400
0626,0627,0629,0630,0631,0633,0634,0635,0671,0672,      00009500
0673,0674,0675,0676,0680,0681,0682,0683,0684,0685,      00009600
0686,0687,0688,0689,0690,0691,0692,0701,0702,0703,      00009700
0705,0706,0707,0709,0710,1711,2711,3711,0713,0714,      00009800
0715,0717,0718,0719,0725,0726,0727,0729,0730,0731,      00009900
0733,0734,0735,0737,0738,0739,0741,0742,0743,0745,      00010000
0746,0747,0749,0750,0751,0753,0754,0755,0771,0772,      00010100
0773,0774,0775,0776,0777,0778,0779,0783,0784,0785,      00010200
0786,0787,0788,0789,0790,0791,0792,0793,0794,0795,      00010300
0796,1803,2803,0805,0806,0807,0809,0810,0811,0813,      00010400
0814,0815,0817,0818,0819,0821,0822,0827,0829,0830,      00010500
0831,0833,0834,0835,0837,0838,0839,0841,0842,0843,      00010600
0845,0846,0847,0849,0850,0851,0853,0854,0855,0856,      00010700
0859,0861,0862,0863,0865,0866,0867,0869,0871,0872,      00010800
0873,0874,0875,0876,0877,0878,0879,0880,0881,0882,      00010900
0883,0884,0885,0886,0890,0891,0892,0893,0894,0895,      00011000
0896,0897,0898,0899,0981,0982,      00011100
0903,0905,0906,0907,0992,0993,      00011200
1302,2302,3302,0338,0547,0567,1607,2607,3607,0614) ;
DCL      HJCHK(572)      PIC '(6)9' INIT(      00011400
111735,112630,112631,111540,112717,111542,111551,112636,112123,112123,      00011500
111223,112611,112534,111639,112025,111417,111412,111513,112028,111631,      00011600
111919,111513,111616,112027,111722,111529,112324,111814,112030,111820,      00011700
112134,111512,112234,111630,111917,112319,112024,112523,112020,112528,      00011800
111116,111716,111543,111222,111919,112331,111515,111728,111228,111627,      00011900
112518,111532,111537,112020,112517,112120,111114,111616,111813,112031,      00012000
111428,111546,111536,111421,111514,111723,112034,111128,112422,112235,      00012100
111227,111437,111120,111550,111228,111728,112212,112525,111823,111629,      00012200
112119,111711,112414,111811,112319,111548,111428,112028,111522,112323,      00012300
111520,111416,111412,112517,112222,111548,112526,111930,111540,111926,      00012400
112635,112411,112725,111630,111636,112022,111620,112613,112528,112514,      00012500
111217,111927,111621,112117,111124,112530,112425,111325,111439,111627,      00012600

```

```

111635,112619,112619,112525,111739,111613,111741,112011,111515,111226, 00012700
112123,111745,111734,112114,112715,112626,112612,111415,111632,112720, 00012800
112629,112314,111121,112714,112725,112615,112714,112616,112626,112626, 00012900
112625,112616,111631,112217,111433,112320,111222,111318,111518,112416, 00013000
112323,112226,111726,112318,111631,111219,111427,112233,111815,112331, 00013100
112314,112718,211227,211827,211228,211828,211828,211818,211424,212021, 00013200
211511,211425,211538,211221,211630,211523,211114,212013,211619,211526, 00013300
211615,212021,211615,211727,211626,211641,211226,211537,211329,211637, 00013400
211732,211321,211644,211623,212017,211617,211516,211411,211614,211822, 00013500
211617,211651,211739,211619,211619, 00013600
211535,211629,211733,211826,211832,211725,211637,211712,211636,211922, 00013700
211223,212013,211533,211617,211819,211424,211415,211624,211526,211718, 00013800
211817,211716,211424,221628,221235,221222,221325,221225,221324,221331, 00013900
221320,221412,221215,221514,221211,221318,221217,221128,221523,221521, 00014000
221611,221214,221126,221319,221328,221523,221411,221326,221129,221217, 00014100
221215,221331,221315,221516,221417,221516,221323,221211,221334,221521, 00014200
221514,231219,231423,231432,231319,231415,231220,231115,231119,231113, 00014300
231318,231313,231313,231328,231413,231322,231417,231219,231212,231326, 00014400
231427,231318,231424,231419,231316,311712,311714,311114,311127,311120, 00014500
311115,311126,311237,311218,311239,311218,311213,311211, 00014600
311227,311220,311226,311320,311319,311313,311424,311413,311413,311414, 00014700
311412,311424,311520,311520,311520,311520,311511,311518,311513,311513, 00014800
311615,311614,311611,311620,311411,321127,321117,321111,321219,321220, 00014900
321223,321416,321420,321525,321417,321321,321322,321314,321612,313412, 00015000
313912,313712,314511,314511,313411,314536,314434,314737,313537,314633, 00015100
314137,314339,315033,315033,315033,313631,324411,323712,323311,323732, 00015200
323133,323432,321622,331128,331116,331118,331125,331218,331213,331314, 00015300
331313,340115,340222,340112,340129,340126,340214,340117,340224,340231, 00015400
340212,340231, 00015500
340245,341120,341112,333411,333931,333831,333636,334034,333138,344112, 00015600
343211,343111,343811,343536,344331,343736,344541,344433,344242,344032, 00015700
343137,343432,351141,351135,351127,351120,351117,351123,351126,351418, 00015800
351226,351226,351226,351412,351511,351217,351415,351312,351216,360227, 00015900
360115,360211,360122,260317,360112,360332,360220,360213,360119,360121, 00016000
360314,360119, 360324,361113,361137,361132,361117, 00016100
361122,361234,361316,361225,361213,361112,353411,353112,354334,353633, 00016200
354234,354035,353135,353841,354136,364011,364811,363211,364211,363731, 00016300
363938,364934,364133,364531,363134,363536,364641,364432,364042,371121, 00016400
371121,371134,371114,371123,371111,371223,371228,371211,371319,371519, 00016500
371320,371314,371423,371414,371420,381111,381119,381117,381120,381135, 00016600
381134,381116,381516,381143,381128,381142,381242,381230,381227,381213, 00016700
381240,381225,381216,381215,381818,381820,381423,381331,381325,381323, 00016800
381327,381612,381618,381617,381423,381427,374913,374811,373411,375311, 00016900
373912,374332,374734,374432,373343,374536,373731,373440,374034,374634, 00017000
375131,374032,383412,383612,383735,384431,384834,383334,384737,384242, 00017100
384336,383836,383146,384537,391119,391114,391125,391114,391116,393111,393131, 00017200
211651,211651,211651,212018,311423,321122,331125,331125,331125,331121 00017300
);
DCL ONCODE BUILTIN ;
ON KEY(TPO) BEGIN ;
IF ONCODE = 52 THEN DO ;
    PUT SKIP EDIT ('DATA DUPLICATED','KEY=',SUBSTR(TT,7,14),TT) 00017400
    (A,A,A,X(2),A(100)); 00017500
    ECNT = ECNT + 1 ; 00017600
GO TO RD; END ; 00017700
END; 00017800
ON ENDFILE(TPI) GO TO LAST ; 00017900
OPEN FILE(TPI),FILE(TPO) DIRECT UPDATE; 00018000
RD: READ FILE(TPI) INTO(TT) ; 00018100
ICNT = ICNT + 1 ; 00018200
TTT=''; 00018300
IF SUBSTR(TT,19,2)='00' THEN 00018400
DO ; 00018500
00018600
00018700
00018800
00018900

```



```

CALL HGBATCH(FUNC,RECODE,ADDR,LA,DDD,LB) ; 00019000
QADDR = DDD; 00019100
TAA = TA , BY NAME ; 00019200
TAA.BIGO = 'S' ; 00019300
DO I=1 TO 572; 00019400
    IF TA.JOSA=DJOSA(I) THEN HJ = HJCHK(I) ; 00019500
END; 00019600
END ; 00019700
ELSE DO; 00019800
    CALL HGBATCH(FUNC,RECODE,NAME,LC,ABC,LD) ; 00019900
    ONAME = ABC ; 00020000
    TBB = TB , BY NAME ; 00020100
    TBB.BIGOA = 'S' ; 00020200
END; 00020300
IF RECODE = ' ' THEN DO ; 00020400
    PUT SKIP(2) EDIT('HAN ERROR : ',RECODE,STRING(TT)) 00020500
    (A,X(2)) ; 00020600
    ECNT = ECNT + 1 ; 00020700
    GOTO RD ; 00020800
END ; 00020900
WRITE FILE(TPO) FROM(TTT) KEYFROM(SUBSTR(TTT,7,14)) ; 00021000
OCNT = OCNT + 1 ; 00021100
GOTO RD ; 00021200
LAST: CLOSE FILE(TPO),FILE(TPI) ; 00021300
PUT SKIP EDIT('INSERT GAGU CNT = ',ICNT)(A,F(5)); 00021400
PUT SKIP EDIT('ERROR GAGU CNT = ',ECNT)(A,F(5)); 00021500
PUT SKIP EDIT('OUTPUT GAGU CNT = ',OCNT)(A,F(5)); 00021600
END KANG ; 00021700
/* 00021800
//GO.TPI DD DSN=A33.VH.TEM(SVHS0893),UNIT=DISK,VOL=SER=BOS005, 00021900
// DCB=(RECFM=FB,LRECL=100,BLKSIZE=4000),DISP=OLD 00022000
//GO.TPO DD DSN=A33.VH.MS,DISP=SHR 00022100
//GO.SYSPRINT DD SYSOUT=* 00022200
// 00022300

```

```

00000000
00100000
00200000
00300000
00400000
00500000
00600000
00700000
00800000
00900000

```

A33.86.055,17:45:03,A33.PROGRAM.LIBRARY(CHGMATCH)

```
//A33VHWM2 JOB CLASS=A,MSGCLASS=X,USER=A33,PASSWD=PINE
//STEP2 EXEC PLIFCLG,REGION=1024K
//PLI.SYSIN DD *
* PROCESS A,S,GS;
  HONG: PROCEDURE OPTIONS(MAIN);
    DCL SIN FILE RECORD INPUT,
        CHANGE FILE RECORD INPUT ;
    DCL 1 INR1,
        2 FIL CHAR(6),
        2 JOSAGU1 CHAR(4),
        2 BLOCK1 CHAR(2), /*CHANGE*/
        2 GUBUN CHAR(1),
        2 FILA CHAR(67);
    DCL 1 INR2,
        2 FILA CHAR(6),
        2 JOSAGU2 CHAR(4), /* SIN */
        2 BLOCK2 CHAR(2),
        2 GAGU_NO CHAR(8),
        2 FILB CHAR(60) ;
    DCL (CHK,SAVE) CHAR(6) INIT('0');
    DCL CNT PIC'99999' INIT(0);
    DCL (EOF1,EOF2) BIT(1),
        YES BIT(1) INIT('1'B),
        NO BIT(1) INIT('0'B);
    OPEN FILE(SIN),FILE(CHANGE),
        FILE(SYSPRINT) PAGESIZE(130);
    ON ENDFILE(CHANGE) EOF1 = YES;
    ON ENDFILE(SIN) EOF2 = YES;
    CALL TITLE_RTN;
    CALL HEAD_RTN;
    EOF1 = NO;
    EOF2 = NO;
    READ FILE(CHANGE) INTO(INR1);
    READ FILE(SIN) INTO(INR2);
    SAVE = JOSAGU2 || BLOCK2;
    DO WHILE(EOF1 = NO );
      IF GUBUN='S' THEN
        DO;
          CHK = JOSAGU1 || BLOCK1;
          CALL FIRST_RTN;
        END;
      ELSE;
        READ FILE(CHANGE) INTO(INR1);
      END;
    PUT SKIP(2) EDIT('COUNT RECORD =',CNT)(A,F(9));
    CLOSE FILE(SIN),FILE(CHANGE),FILE(SYSPRINT);
    /*****/
    FIRST_RTN: PROC;
      DO WHILE( EOF2 = NO & GUBUN='S' & CHK<>SAVE);
        IF CHK=SAVE THEN DO;
          CALL OK_RTN;
          READ FILE(SIN) INTO(INR2);
          SAVE = JOSAGU2 || BLOCK2;
        END;
      ELSE IF CHK>SAVE THEN DO;
          CALL ERROR_RTN;
          READ FILE(SIN) INTO(INR2);
          SAVE = JOSAGU2 || BLOCK2;
        END;
      ELSE GUBUN = ' ';
    END;
  END FIRST_RTN;
  /*****/
```

```

TITLE_RTN: PROC;                                00006400
PUT SKIP(3) EDIT                                00006500
('***** GAGU SAMPLE MANAGEMENT ERROR_CHECK (SIN GUE) *****') 00006600
(X(10),A(58));                                  00006700
END TITLE_RTN;                                  00006800
/*****/                                          00006900
HEAD_RTN: PROC;                                  00007000
PUT SKIP(2) EDIT                                00007100
('GEOCHE_NAIYUCK', 'SINGUE_GAGU & GAGU_NO', 'CHECK_GUBUN') 00007200
(X(5),A(14),X(1),A(21),X(4),A(11));            00007300
END HEAD_RTN;                                    00007400
/*****/                                          00007500
ERROR_RTN: PROC;                                00007600
CNT = CNT + 1;                                  00007700
PUT SKIP(2) EDIT(CHK,SAVE,GAGU_NO, '*****') 00007800
(X(8),A(6),X(10),A(6),X(4),A(8),X(5),A(5));    00007900
END ERROR_RTN;                                  00008000
/*****/                                          00008100
OK_RTN: PROC;                                    00008200
CNT = CNT + 1;                                  00008300
/* PUT SKIP(2) EDIT(CHK,SAVE,GAGU_NO, 'OK') 00008400
(X(8),A(6),X(10),A(6),X(4),A(8),X(5),A(3)); */ 00008500
END OK_RTN;                                      00008600
/*****/                                          00008700
END HONG;                                        00008800
/*                                                00008900
//GO.SIN DD DSN=A33.PROGRAM.LIBRARY(SIN06),DISP=(OLD,KEEP) 00009000
//GO.CHANGE DD DSN=A33.PROGRAM.LIBRARY(J8506),DISP=(OLD,KEEP) 00009100
//GO.SYSPRINT DD SYSOUT=*                        00009200
//GO.SYSOUT DD SYSOUT=*                          00009300
//                                                00009400

```

A33,86.055,17:42:39,A33.PROGRAM.LIBRARY(SVHUSE93)

```
//A33VHUSE JOB CLASS=B,MSGCLASS=X,USER=A33,PASSWORD=PINE 00000100
//SORT EXEC PGM=SORT,PARM='SIZE(MAX)' 00000200
//SYSOUT DD SYSOUT=* 00000300
//SYSPRINT DD SYSOUT=* 00000400
//SORTIN DD DSN=H01.PROGRAM.LIBRARY(J8505),DISP=SHR 00000500
//SORTOUT DD DSN=A33.PROGRAM.LIBRARY(DATAN),DISP=SHR 00000600
//SORTWK01 DD UNIT=DISK,SPACE=(CYL,(30)),VOL=SER=BOSSTR 00000700
//SYSIN DD * 00000800
SORT FIELDS=(7,6,A),FORMAT=BI,FILSZ=E120000 00000900
/* 00001000
// 00001100
```

A33,86.055,17:50:22,A33.PROGRAM.LIBRARY(SVHWVH93)

```
//A33VHVVH JOB CLASS=A,MSGCLASS=X,NOTIFY=A33          00000100
//JOSATAB EXEC PLIFCLG                                00000200
//PLI.SYSIN DD *                                      00000300
* PROCESS GS,NEST,OPT(2) ;                             00000400
  LEE :                                                 00000500
    PROC OPTIONS(MAIN) ;                                00000600
    DCL TPI FILE RECORD INPUT ;                          00000700
    DCL 1 TA ,                                           00000800
      2 ILR CHAR (18),                                  00000900
      2 CARD CHAR (1),                                  00001000
      2 GITA CHAR (81) ;                                  00001100
    DCL 1 CD1 DEF GITA,                                   00001200
      2 MNAME CHAR(20),                                  00001300
      2 MNO CHAR (2),                                   00001400
      2 CNAME CHAR(20),                                  00001500
      2 BIRF CHAR(1),                                   00001600
      2 BIRY PIC '9',                                   00001700
      2 BIRM PIC '(2)9',                                 00001800
      2 BIRD PIC '(2)9',                                 00001900
      2 SEX CHAR (1),                                   00002000
      2 MAGE CHAR (2),                                  00002100
      2 SEQ CHAR (2),                                   00002200
      2 IMG CHAR (2),                                   00002300
      2 MED CHAR (2),                                   00002400
      2 MEDF CHAR (1),                                  00002500
      2 JOSAN CHAR(1),                                  00002600
      2 BLK1 CHAR (22) ;                                 00002700
    DCL 1 CD2 DEF GITA,                                   00002800
      2 SNAME CHAR(20),                                  00002900
      2 GAWON CHAR(2),                                   00003000
      2 SSEX CHAR (1),                                  00003100
      2 BIY PIC '(3)9',                                  00003200
      2 BIM PIC '(2)9',                                  00003300
      2 BID PIC '(2)9',                                  00003400
      2 SBIY PIC '(3)9',                                 00003500
      2 SBIM PIC '(2)9',                                 00003600
      2 SBID PIC '(2)9',                                 00003700
      2 SED CHAR (2),                                   00003800
      2 SEDF CHAR (1),                                  00003900
      2 JKU CHAR (2) ;                                   00004000
      2 HONS CHAR(1),                                   00004100
      2 JIN CHAR (1),                                   00004200
      2 SAWO CHAR(2),                                   00004300
      2 BLK2 CHAR (35);                                 00004400
    DCL 1 CA3 DEF GITA,                                   00004500
      2 BNAME CHAR(20),                                  00004600
      2 BGNO CHAR (2),                                   00004700
      2 BAGE CHAR (2),                                   00004800
      2 BED CHAR (2),                                   00004900
      2 BEDF CHAR (1),                                  00005000
      2 CNAME CHAR(20),                                  00005100
      2 CHNO CHAR (2),                                   00005200
      2 CHAGE CHAR(2),                                  00005300
      2 CHNN CHAR (2),                                   00005400
      2 CHNF CHAR (1),                                  00005500
      2 HOF PIC '9',                                    00005600
      2 HOY PIC '9',                                    00005700
      2 HOM PIC '(2)9',                                 00005800
      2 BLK3 CHAR (23) ;                                 00005900
    DCL 1 CA4 DEF GITA,                                   00006000
      2 BUNAME CHAR(20),                                 00006100
      2 BUAGE CHAR(2),                                  00006200
      2 CHNAME CHAR(20),                                00006300
```

```

2 CHEAGE CHAR(2), 00006400
2 JASU CHAR(2), 00006500
2 HONY PIC '(2)9', 00006600
2 HONM PIC '(2)9', 00006700
2 BALY PIC '(2)9', 00006800
2 BALM PIC '(2)9', 00006900
2 BLK4 CHAR(27); 00007000
DCL SW CHAR(1) INIT('0'); 00007100
DCL BBSS PIC '(3)9' INIT(0); 00007200
DCL BIA PIC '(3)9' DEF GITA POS(4); 00007300
DCL SBA PIC '(3)9' DEF GITA POS(11); 00007400
DCL JIK(10) CHAR(1) INIT('0','1','2','3','4','5','6','7',
'8','9'); 00007500
DCL JJA(7) PIC '9' INIT(0,1,2,3,4,5,6); 00007700
DCL YIL(20) PIC '(3)9' INIT(001,002,003,004,005,006,013,020,
027,059,060,090,120,150,180,210,240,270,300,330); 00007800
DCL JJ(4) CHAR(1) INIT('1','2','3','9'); 00008000
DCL SS(12) CHAR(2) INIT('11','21','29','31','39','41','49',
'51','61','71','81','99'); 00008100
DCL BBB(5) CHAR(1) INIT('4','1','2','3','9'); 00008200
DCL JOJ(3) CHAR(1) INIT('1','2','3'); 00008300
DCL TAB6(4,3,13) FLOAT DEC(9) INIT((156)0); 00008400
DCL TAB7(4,10,3) FLOAT DEC(9) INIT((120)0); 00008500
DCL TAB8(4,11,12) FLOAT DEC(9) INIT((528)0); 00008600
DCL TAB9(4,10,7) FLOAT DEC(9) INIT((280)0); 00008700
DCL TAB10(4,3,13) FLOAT DEC(9) INIT((156)0); 00008800
DCL TAB11(4,63,7) FLOAT DEC(9) INIT((1764)0); 00008900
DCL TAB12(4,33,13) FLOAT DEC(9) INIT((1716)0); 00009000
DCL TAB13(4,6,5) FLOAT DEC(9) INIT((120)0); 00009100
DCL TAB14(4,54,6) FLOAT DEC(9) INIT((1296)0); 00009200
DCL TAB15(4,63,13) FLOAT DEC(9) INIT((3276)0); 00009300
DCL TAB16(4,25,6) FLOAT DEC(9) INIT((600)0); 00009400
DCL TAB17(4,8,4) FLOAT DEC(9) INIT((128)0); 00009500
DCL TAB18(4,3,13) FLOAT DEC(9) INIT((156)0); 00009600
DCL TAB20(4,3,13) FLOAT DEC(9) INIT((156)0); 00009700
DCL CODE5(63) CHAR(7) INIT(' TOTAL', '1 -', '1 - 4',
'5 - 9', '10 - 14', '15 - 19', '20 - 24', '25 - 29',
'30 - 34', '35 - 39', '40 - 44', '45 - 49', '50 - 54',
'55 - 59', '60 - 64', '65 - 69', '70 - 74', '75 - 79',
'80 - 84', '85 -', 'UNKNOWN'); 00009800
DCL CODE5(63) CHAR(7) INIT(' TOTAL', '1 -', '1 - 4',
'5 - 9', '10 - 14', '15 - 19', '20 - 24', '25 - 29',
'30 - 34', '35 - 39', '40 - 44', '45 - 49', '50 - 54',
'55 - 59', '60 - 64', '65 - 69', '70 - 74', '75 - 79',
'80 - 84', '85 -', 'UNKNOWN'); 00009900
DCL CODE5(63) CHAR(7) INIT(' TOTAL', '1 -', '1 - 4',
'5 - 9', '10 - 14', '15 - 19', '20 - 24', '25 - 29',
'30 - 34', '35 - 39', '40 - 44', '45 - 49', '50 - 54',
'55 - 59', '60 - 64', '65 - 69', '70 - 74', '75 - 79',
'80 - 84', '85 -', 'UNKNOWN'); 00010000
DCL CODE5(63) CHAR(7) INIT(' TOTAL', '1 -', '1 - 4',
'5 - 9', '10 - 14', '15 - 19', '20 - 24', '25 - 29',
'30 - 34', '35 - 39', '40 - 44', '45 - 49', '50 - 54',
'55 - 59', '60 - 64', '65 - 69', '70 - 74', '75 - 79',
'80 - 84', '85 -', 'UNKNOWN'); 00010100
DCL CODE5(63) CHAR(7) INIT(' TOTAL', '1 -', '1 - 4',
'5 - 9', '10 - 14', '15 - 19', '20 - 24', '25 - 29',
'30 - 34', '35 - 39', '40 - 44', '45 - 49', '50 - 54',
'55 - 59', '60 - 64', '65 - 69', '70 - 74', '75 - 79',
'80 - 84', '85 -', 'UNKNOWN'); 00010200
DCL CODE5(63) CHAR(7) INIT(' TOTAL', '1 -', '1 - 4',
'5 - 9', '10 - 14', '15 - 19', '20 - 24', '25 - 29',
'30 - 34', '35 - 39', '40 - 44', '45 - 49', '50 - 54',
'55 - 59', '60 - 64', '65 - 69', '70 - 74', '75 - 79',
'80 - 84', '85 -', 'UNKNOWN'); 00010300
DCL CODE5(63) CHAR(7) INIT(' TOTAL', '1 -', '1 - 4',
'5 - 9', '10 - 14', '15 - 19', '20 - 24', '25 - 29',
'30 - 34', '35 - 39', '40 - 44', '45 - 49', '50 - 54',
'55 - 59', '60 - 64', '65 - 69', '70 - 74', '75 - 79',
'80 - 84', '85 -', 'UNKNOWN'); 00010400
DCL CODE5(63) CHAR(7) INIT(' TOTAL', '1 -', '1 - 4',
'5 - 9', '10 - 14', '15 - 19', '20 - 24', '25 - 29',
'30 - 34', '35 - 39', '40 - 44', '45 - 49', '50 - 54',
'55 - 59', '60 - 64', '65 - 69', '70 - 74', '75 - 79',
'80 - 84', '85 -', 'UNKNOWN'); 00010500
DCL CODE5(63) CHAR(7) INIT(' TOTAL', '1 -', '1 - 4',
'5 - 9', '10 - 14', '15 - 19', '20 - 24', '25 - 29',
'30 - 34', '35 - 39', '40 - 44', '45 - 49', '50 - 54',
'55 - 59', '60 - 64', '65 - 69', '70 - 74', '75 - 79',
'80 - 84', '85 -', 'UNKNOWN'); 00010600
DCL CODE5(63) CHAR(7) INIT(' TOTAL', '1 -', '1 - 4',
'5 - 9', '10 - 14', '15 - 19', '20 - 24', '25 - 29',
'30 - 34', '35 - 39', '40 - 44', '45 - 49', '50 - 54',
'55 - 59', '60 - 64', '65 - 69', '70 - 74', '75 - 79',
'80 - 84', '85 -', 'UNKNOWN'); 00010700
DCL CODE5(63) CHAR(7) INIT(' TOTAL', '1 -', '1 - 4',
'5 - 9', '10 - 14', '15 - 19', '20 - 24', '25 - 29',
'30 - 34', '35 - 39', '40 - 44', '45 - 49', '50 - 54',
'55 - 59', '60 - 64', '65 - 69', '70 - 74', '75 - 79',
'80 - 84', '85 -', 'UNKNOWN'); 00010800
DCL CODE5(63) CHAR(7) INIT(' TOTAL', '1 -', '1 - 4',
'5 - 9', '10 - 14', '15 - 19', '20 - 24', '25 - 29',
'30 - 34', '35 - 39', '40 - 44', '45 - 49', '50 - 54',
'55 - 59', '60 - 64', '65 - 69', '70 - 74', '75 - 79',
'80 - 84', '85 -', 'UNKNOWN'); 00010900
DCL CODE5(63) CHAR(7) INIT(' TOTAL', '1 -', '1 - 4',
'5 - 9', '10 - 14', '15 - 19', '20 - 24', '25 - 29',
'30 - 34', '35 - 39', '40 - 44', '45 - 49', '50 - 54',
'55 - 59', '60 - 64', '65 - 69', '70 - 74', '75 - 79',
'80 - 84', '85 -', 'UNKNOWN'); 00011000
DCL CODE5(63) CHAR(7) INIT(' TOTAL', '1 -', '1 - 4',
'5 - 9', '10 - 14', '15 - 19', '20 - 24', '25 - 29',
'30 - 34', '35 - 39', '40 - 44', '45 - 49', '50 - 54',
'55 - 59', '60 - 64', '65 - 69', '70 - 74', '75 - 79',
'80 - 84', '85 -', 'UNKNOWN'); 00011100
DCL CODE5(63) CHAR(7) INIT(' TOTAL', '1 -', '1 - 4',
'5 - 9', '10 - 14', '15 - 19', '20 - 24', '25 - 29',
'30 - 34', '35 - 39', '40 - 44', '45 - 49', '50 - 54',
'55 - 59', '60 - 64', '65 - 69', '70 - 74', '75 - 79',
'80 - 84', '85 -', 'UNKNOWN'); 00011200
DCL CODE5(63) CHAR(7) INIT(' TOTAL', '1 -', '1 - 4',
'5 - 9', '10 - 14', '15 - 19', '20 - 24', '25 - 29',
'30 - 34', '35 - 39', '40 - 44', '45 - 49', '50 - 54',
'55 - 59', '60 - 64', '65 - 69', '70 - 74', '75 - 79',
'80 - 84', '85 -', 'UNKNOWN'); 00011300
DCL CODE12(33) CHAR(10) INIT(' TOTAL TOT', '0',
'1', '2', '3', '4',
'5', '6', '7', '8',
'9', 'MALE TOT', '0', '1',
'2', '3', '4', '5',
'6', '7', '8', '9',
'FEMALE TOT', '0', '1', '2',
'3', '4', '5', '6',
'7', '8', '9'); 00011400
DCL TIT12 CHAR(91) INIT(' TOTAL 15- 15-19 20-24 25-29
30-34 35-39 40-44 45-49 50-54 55-59 60-64 65+'); 00011500
DCL CODE14(54) CHAR(7) INIT('T-TOTAL', '15 -',
'15 - 19', '20 - 24', '25 - 29', 00011600
00011700
00011800
00011900
00012000
00012100
00012200
00012300
00012400
00012500
00012600

```

```

'30 - 34', '35 - 39', '40 - 44', '45 - 49', '50 - 54', 00012700
'55 - 59', '60 - 64', '65 - 69', '70 - 74', '75 - 79', 00012800
'80 - 84', '85 +', 'UN', 'M-TOTAL', '15 -', 00012900
'15 - 19', '20 - 24', '25 - 29', 00013000
'30 - 34', '35 - 39', '40 - 44', '45 - 49', '50 - 54', 00013100
'55 - 59', '60 - 64', '65 - 69', '70 - 74', '75 - 79', 00013200
'80 - 84', '85 +', 'UN', 'F-TOTAL', '15 -', 00013300
'15 - 19', '20 - 24', '25 - 29', 00013400
'30 - 34', '35 - 39', '40 - 44', '45 - 49', '50 - 54', 00013500
'55 - 59', '60 - 64', '65 - 69', '70 - 74', '75 - 79', 00013600
'80 - 84', '85 +', 'UN', 00013700
DCL CTIT(13) CHAR(2) INIT('15', '20', '21', '22', '23', '24', '25', 00013800
'26', '27', '28', '29', '30', '35');
DCL BTIT(14) CHAR(2) INIT('15', '20', '25', '26', '27', '28', '29', 00014000
'30', '31', '32', '33', '34', '35', '40');
DCL TIT14 CHAR(60) INIT(' TOTAL 4 1 00014200
3 9');
DCL CODE13(6) CHAR(10) INIT(' TOTAL', ' 11', 00014400
' 88', ' TOTAL', ' MALE', ' FEMALE');
DCL TIT13 CHAR(50) INIT(' TOTAL 1 2 00014600
9');
DCL TIT15 CHAR(91) INIT(' TOTAL 11 21 29 31 00014800
41 49 51 61 71 81 99');
DCL CODE16(25) CHAR(7) INIT(' TOTAL', ' 1 -', ' 1-6', 00015000
' 1', ' 2', ' 3', ' 4', ' 5',
' 6', ' 7-27', ' 7-13', ' 14-20', ' 21-27',
' 28-2M', ' 2M-11M', ' 2M', ' 3M', ' 4M',
' 5M', ' 6M', ' 7M', ' 8M', ' 9M',
' 10M', ' 11M');
DCL TIT16 CHAR(60) INIT(' TOTAL MALE FEMALE 00015600
MALE FEMALE');
DCL MONTH(12) PIC '(2)9' INIT(01,02,03,04,05,06,07, 00015800
08,09,10,11,12);
DCL Y1(12) CHAR(2) INIT('14', '19', '24', '29', '34', '39', '44', 00016000
'49', '54', '59', '64');
DCL Y2(18) CHAR(2) INIT('00', '04', '09', '14', '19', '24', '29', 00016200
'34', '39', '44', '49', '54', '59', '64',
'69', '74', '79', '84');
DCL MXX CHAR(3) INIT('0'); 00016500
DCL YYY(18) PIC '(3)9' INIT(984,979,974,969,964,959,954,949, 00016600
944,939,934,929,924,919,914,909,904,899);
DCL CODE1(4) CHAR(13) INIT('WHOLE COUNTRY', 'GREAT CITY', 00016800
'SI : BU', 'GUN : BU');
DCL CODE2 CHAR(91) INIT(' TOTAL 1 2 3 00017000
6 7 8 9 10 11 12');
DCL CODE3(3) CHAR(6) INIT(' TOTAL', ' MALE', ' FEMALE'); 00017200
DCL CODE4(4) CHAR(8) INIT('TABLE #6', 'TABLE 18', 00017300
'TABLE 20', 'TABLE 10');
DCL CODE6 CHAR(70) INIT(' TOTAL NO-SCHOOL P-SCHOOL M-00017500
H-SCHOOL COLLAGE + UNKNOWN');
DCL CODE7(11) CHAR(7) INIT(' TOTAL', ' 15 -', '15 - 19', 00017700
'20 - 24', '25 - 29', '30 - 34', '35 - 39',
'40 - 44', '45 - 49', '50 +', 'UNKNOWN');
DCL CODE8 CHAR(30) INIT(' TOTAL MALE FEMALE'); 00018000
DCL Y3(8) CHAR(2) INIT('14', '19', '24', '29', '34', '39', 00018100
'44', '49');
DCL KA1(9) CHAR(2) INIT('01', '02', '03', '04', '05', 00018300
'06', '07', '08', '09');
DCL KA2(4) CHAR(2) INIT('00', '06', '09', '12'); 00018500
DCL COD1 CHAR(96) INIT(' TOTAL 1 2 3 00018600
5 6 7 8 9 10 + UN');
OPEN FILE(TPI), FILE(SYSPRINT) LINESIZE(132); 00018800
ON ENDFILE(TPI) GO TO LAST; 00018900

```

PUT PAGE ;	00019000
RD : READ FILE(TPI) INTO(TA) ;	00019100
IF CARD = '1' THEN GO TO CARD1 ;	00019200
IF CARD = '2' THEN GO TO CARD2 ;	00019300
IF CARD = '3' THEN GO TO CARD3 ;	00019400
IF CARD = '4' THEN GO TO CARD4 ;	00019500
GO TO RD ;	00019600
CARD1 :	00019700
IF BIRY = 4 THEN GO TO RD ;	00019800
IF BIRM > 9 THEN GO TO RD ;	00019900
IF SUBSTR(ILR,8,3) > '500' THEN GO TO A1 ;	00020000
N = 2 ;CALL BAA ;	00020100
GO TO RD ;	00020200
A1 : IF SUBSTR(ILR,9,1) >= '7' THEN GO TO A11 ;	00020300
N = 3 ;CALL BAA ;	00020400
GO TO RD ;	00020500
A11 : N = 4 ;CALL BAA ;GO TO RD ;	00020600
CARD2 :	00020700
IF SBIY = 984 THEN GO TO RD ;	00020800
IF SBIM > 9 THEN GO TO RD ;	00020900
IF SUBSTR(ILR,8,3) > '500' THEN GO TO B1 ;	00021000
N = 2 ;CALL BAB ;	00021100
GO TO RD ;	00021200
B1 : IF SUBSTR(ILR,9,1) >= '7' THEN GO TO B11 ;	00021300
N = 3 ;CALL BAB ;	00021400
GO TO RD ;	00021500
B11 : N = 4 ;CALL BAB ;	00021600
GO TO RD ;	00021700
CARD3 :	00021800
IF HOY = 4 THEN GO TO RD ;	00021900
IF HOM > 9 THEN GO TO RD ;	00022000
IF SUBSTR(ILR,8,3) > '500' THEN GO TO D1 ;	00022100
N = 2 ; GO TO DD ;	00022200
D1 : IF SUBSTR(ILR,9,1) >= '7' THEN GO TO D2 ;	00022300
N = 3 ; GO TO DD ;	00022400
D2 : N = 4 ;	00022500
DD : DO L = 1 TO 12 ;	00022600
IF HOM = MONTH(L) THEN GO TO D3 ;	00022700
END ;	00022800
GO TO RD ;	00022900
D3 : TAB18(N,3,L+1) = TAB18(N,3,L+1) + 1 ;	00023000
GO TO RD ;	00023100
CARD4 :	00023200
IF Baly > 84 THEN GO TO RD ;	00023300
IF BALM > 9 THEN GO TO RD ;	00023400
IF SUBSTR(ILR,8,3) > '500' THEN GO TO F1 ;	00023500
N = 2 ; GO TO DDK ;	00023600
F1 : IF SUBSTR(ILR,9,1) >= '7' THEN GO TO F2 ;	00023700
N = 3 ; GO TO DDK ;	00023800
F2 : N = 4 ;	00023900
DDK :	00024000
DO L = 1 TO 12 ;	00024100
IF BALM = MONTH(L) THEN GO TO D4 ;	00024200
END ;	00024300
GO TO RD ;	00024400
D4 : TAB20(N,3,L+1) = TAB20(N,3,L+1) + 1 ;	00024500
GO TO RD ;	00024600
LAST : TAB6(*,1,*) = TAB6(*,2,*) + TAB6(*,3,*) ;	00024700
DO M = 2 TO 11 ;	00024800
TAB8(*,1,*) = TAB8(*,1,*) + TAB8(*,M,*) ;	00024900
END ;	00025000
DO M = 2 TO 5 ;	00025100
TAB13(*,*,1) = TAB13(*,*,1) + TAB13(*,*,M) ;	00025200



```

END ;
TAB13(*,4,*) = TAB13(*,5,*) + TAB13(*,6,*) ;
TAB13(*,1,*) = TAB13(*,2,*) + TAB13(*,3,*) ;
TAB16(*,*,1) = TAB16(*,*,2) + TAB16(*,*,3) ;
TAB16(*,*,4) = TAB16(*,*,5) + TAB16(*,*,6) ;
DO M = 2 TO 10 ;
TAB7(*,1,*) = TAB7(*,1,*) + TAB7(*,M,*) ;
END ;
DO I = 2 TO 10 ;
TAB9(*,1,*) = TAB9(*,1,*) + TAB9(*,I,*) ;
END ;
DO M = 2 TO 6 ;
TAB14(*,*,1) = TAB14(*,*,1) + TAB14(*,*,M) ;
END ;
DO M = 2 TO 12 ;
TAB8(*,*,1) = TAB8(*,*,1) + TAB8(*,*,M) ;
END ;
DO M = 1 TO 21 ;
TAB11(*,M,*) = TAB11(*,M+21,*) + TAB11(*,M+42,*) ;
TAB15(*,M,*) = TAB15(*,M+21,*) + TAB15(*,M+42,*) ;
END ;
DO M = 1 TO 18 ;
TAB14(*,M,*) = TAB14(*,M+18,*) + TAB14(*,M+36,*) ;
END ;
DO M = 1 TO 11 ;
TAB12(*,M,*) = TAB12(*,M+11,*) + TAB12(*,M+22,*) ;
END ;
DO L = 2 TO 7 ;
TAB9(*,*,1) = TAB9(*,*,1) + TAB9(*,*,L) ;
TAB11(*,*,1) = TAB11(*,*,1) + TAB11(*,*,L) ;
END ;
DO M = 2 TO 21 ;
TAB11(*,1,*) = TAB11(*,1,*) + TAB11(*,M,*) ;
TAB15(*,1,*) = TAB15(*,1,*) + TAB15(*,M,*) ;
END ;
DO M = 23 TO 42 ;
TAB11(*,22,*) = TAB11(*,22,*)+TAB11(*,M,*) ;
TAB15(*,22,*) = TAB15(*,22,*)+TAB15(*,M,*) ;
END ;
DO M = 44 TO 63 ;
TAB11(*,43,*) = TAB11(*,43,*)+TAB11(*,M,*) ;
TAB15(*,43,*) = TAB15(*,43,*)+TAB15(*,M,*) ;
END ;
DO M = 2 TO 11 ;
TAB12(*,1,*) = TAB12(*,1,*) + TAB12(*,M,*) ;
END ;
DO M = 13 TO 22 ;
TAB12(*,12,*) = TAB12(*,12,*)+TAB12(*,M,*) ;
END ;
DO M = 24 TO 33 ;
TAB12(*,23,*) = TAB12(*,23,*)+TAB12(*,M,*) ;
END ;
DO M = 2 TO 18 ;
TAB14(*,1,*) = TAB14(*,1,*) + TAB14(*,M,*) ;
END ;
DO M = 20 TO 36 ;
TAB14(*,19,*) = TAB14(*,19,*)+TAB14(*,M,*) ;
END ;
DO M = 38 TO 54 ;
TAB14(*,37,*) = TAB14(*,37,*)+TAB14(*,M,*) ;
END ;
TAB10(*,1,*) = TAB10(*,2,*) + TAB10(*,3,*) ;
TAB18(*,1,*) = TAB18(*,2,*) + TAB18(*,3,*) ;

```

```

00025300
00025400
00025500
00025600
00025700
00025800
00025900
00026000
00026100
00026200
00026300
00026400
00026500
00026600
00026700
00026800
00026900
00027000
00027100
00027200
00027300
00027400
00027500
00027600
00027700
00027800
00027900
00028000
00028100
00028200
00028300
00028400
00028500
00028600
00028700
00028800
00028900
00029000
00029100
00029200
00029300
00029400
00029500
00029600
00029700
00029800
00029900
00030000
00030100
00030200
00030300
00030400
00030500
00030600
00030700
00030800
00030900
00031000
00031100
00031200
00031300
00031400
00031500

```

```

    TAB20(*,1,*) = TAB20(*,2,*) + TAB20(*,3,*) ;
DO N = 2 TO 13 ;
    TAB6(*,*,1) = TAB6(*,*,1)+TAB6(*,*,N) ;
    TAB10(*,*,1) = TAB10(*,*,1)+TAB10(*,*,N) ;
    TAB12(*,*,1) = TAB12(*,*,1)+TAB12(*,*,N) ;
    TAB15(*,*,1) = TAB15(*,*,1)+TAB15(*,*,N) ;
    TAB18(*,*,1) = TAB18(*,*,1)+TAB18(*,*,N) ;
    TAB20(*,*,1) = TAB20(*,*,1)+TAB20(*,*,N) ;
END ;
    TAB7(*,*,1) = TAB7(*,*,2) + TAB7(*,*,3) ;
DO M = 4 TO 9 ;
    TAB16(*,3,*) = TAB16(*,3,*) + TAB16(*,M,*) ;
END ;
DO M = 11 TO 13 ;
    TAB16(*,10,*) = TAB16(*,10,*) + TAB16(*,M,*) ;
END ;
DO M = 16 TO 25 ;
    TAB16(*,15,*) = TAB16(*,15,*) + TAB16(*,M,*) ;
END ;
DO J = 2 TO 25 ;
    TAB16(*,1,*) = TAB16(*,1,*) + TAB16(*,J,*) ;
END ;
DO R = 2 TO 4 ;
    TAB6(1,*,*) = TAB6(1,*,*)+TAB6(R,*,*) ;
    TAB7(1,*,*) = TAB7(1,*,*)+TAB7(R,*,*) ;
    TAB8(1,*,*) = TAB8(1,*,*)+TAB8(R,*,*) ;
    TAB9(1,*,*) = TAB9(1,*,*) + TAB9(R,*,*) ;
    TAB10(1,*,*) = TAB10(1,*,*) + TAB10(R,*,*) ;
    TAB11(1,*,*) = TAB11(1,*,*) + TAB11(R,*,*) ;
    TAB12(1,*,*) = TAB12(1,*,*) + TAB12(R,*,*) ;
    TAB13(1,*,*) = TAB13(1,*,*) + TAB13(R,*,*) ;
    TAB14(1,*,*) = TAB14(1,*,*) + TAB14(R,*,*) ;
    TAB15(1,*,*) = TAB15(1,*,*) + TAB15(R,*,*) ;
    TAB16(1,*,*) = TAB16(1,*,*) + TAB16(R,*,*) ;
    TAB18(1,*,*) = TAB18(1,*,*)+ TAB18(R,*,*) ;
    TAB20(1,*,*) = TAB20(1,*,*)+ TAB20(R,*,*) ;
END ;
/***** PRINT *****/
    PUT SKIP(3) EDIT('TABLE #6')(X(5),A(9)) ;
DO M = 1 TO 4 ;
    PUT SKIP(2) EDIT(CODE1(M))(X(8),A(13)) ;
    PUT SKIP(2) EDIT(CODE2)(X(20),A) ;
    PUT SKIP(1) EDIT((91)'-')(X(20),A(91)) ;
DO K = 1 TO 3 ;
    PUT SKIP(2) EDIT(CODE3(K),(TAB6(M,K,L) DO L = 1 TO 13))
        (X(14),A(6),13 F(7)) ;
END ;
    END ; PUT PAGE ;
    PUT SKIP(3) EDIT('TABLE #7')(X(5),A(9)) ;
DO M = 1 TO 4 ;
    PUT SKIP(2) EDIT(CODE1(M))(X(8),A(13)) ;
    PUT SKIP(2) EDIT(CODE8)(X(20),A) ;
    PUT SKIP(1) EDIT((30)'-')(X(20),A(30)) ;
DO K = 1 TO 10 ;
    PUT SKIP(2) EDIT(CODE7(K),(TAB7(M,K,L) DO L = 1 TO 3))
        (X(13),A,7 F(10)) ;
END ;
    END ;
    PUT SKIP(3) EDIT('TABLE #8')(X(5),A(9)) ;
DO M = 1 TO 4 ;
    PUT SKIP(2) EDIT(CODE1(M))(X(8),A(13)) ;
    PUT SKIP(2) EDIT(COD1)(X(20),A) ;
    PUT SKIP(1) EDIT((96)'-')(X(20),A(96)) ;

```

```

00031600
00031700
00031800
00031900
00032000
00032100
00032200
00032300
00032400
00032500
00032600
00032700
00032800
00032900
00033000
00033100
00033200
00033300
00033400
00033500
00033600
00033700
00033800
00033900
00034000
00034100
00034200
00034300
00034400
00034500
00034600
00034700
00034800
00034900
00035000
00035100
00035200
00035300
00035400
00035500
00035600
00035700
00035800
00035900
00036000
00036100
00036200
00036300
00036400
00036500
00036600
00036700
00036800
00036900
00037000
00037100
00037200
00037300
00037400
00037500
00037600
00037700
00037800

```

```

DO K = 1 TO 11 ;
  PUT SKIP(2) EDIT(CODE7(K), (TAB8(M,K,L) DO L = 1 TO 12))
    (X(13),A,12 F(8)) ;
END ;
END ;
  PUT PAGE ;
  PUT SKIP(3) EDIT('TABLE #9')(X(5),A(9)) ;
DO M = 1 TO 4 ;
  PUT SKIP(2) EDIT(CODE1(M))(X(8),A(13)) ;
  PUT SKIP(2) EDIT(CODE6)(X(20),A) ;
  PUT SKIP(1) EDIT((70)'-')(X(20),A(70)) ;
DO K = 1 TO 10 ;
  PUT SKIP(2) EDIT(CODE7(K), (TAB9(M,K,L) DO L = 1 TO 7))
    (X(13),A,7 F(10)) ;
END ; PUT SKIP(3) ;
END ;
  PUT PAGE ;
  PUT SKIP(3) EDIT('TABLE #10')(X(5),A(9)) ;
DO M = 1 TO 4 ;
  PUT SKIP(2) EDIT(CODE1(M))(X(8),A(13)) ;
  PUT SKIP(2) EDIT(CODE2)(X(20),A) ;
  PUT SKIP(1) EDIT((91)'-')(X(20),A(91)) ;
DO K = 1 TO 3 ;
  PUT SKIP(2) EDIT(CODE3(K), (TAB10(M,K,L) DO L=1 TO 13))
    (X(14),A(6),13 F(7)) ;
END ;
END ; PUT PAGE ;
  PUT SKIP(3) EDIT('TABLE #11')(X(5),A(9)) ;
DO N = 1 TO 4 ;
  PUT SKIP(3) EDIT(CODE1(N))(X(8),A(13)) ;
  PUT SKIP(2) EDIT('TOTAL')(X(8),A(5)) ;
  PUT SKIP(2) EDIT(CODE6)(X(20),A) ;
  PUT SKIP(1) EDIT((70)'-')(X(20),A(70)) ;
DO K = 1 TO 21 ;
  PUT SKIP(2) EDIT(CODE5(K), (TAB11(N,K,L) DO L=1 TO 7))
    (X(13),A,7 F(10)) ;
END ; PUT PAGE ;
  PUT SKIP(2) EDIT('MALE')(X(9),A(4)) ;
  PUT SKIP(2) EDIT(CODE6)(X(20),A) ;
  PUT SKIP(1) EDIT((70)'-')(X(20),A(70)) ;
DO K = 22 TO 42 ;
  PUT SKIP(2) EDIT(CODE5(K), (TAB11(N,K,L) DO L=1 TO 7))
    (X(13),A,7 F(10)) ;
END ; PUT PAGE ;
  PUT SKIP(2) EDIT('FEMALE')(X(7),A(6)) ;
  PUT SKIP(2) EDIT(CODE6)(X(20),A) ;
  PUT SKIP(1) EDIT((70)'-')(X(20),A(70)) ;
DO K = 43 TO 63 ;
  PUT SKIP(2) EDIT(CODE5(K), (TAB11(N,K,L) DO L=1 TO 7))
    (X(13),A,7 F(10)) ;
END ; PUT PAGE ;
END ;
  PUT SKIP(3) EDIT('TABLE #18')(X(5),A(9)) ;
DO M = 1 TO 4 ;
  PUT SKIP(2) EDIT(CODE1(M))(X(8),A(13)) ;
  PUT SKIP(2) EDIT(CODE2)(X(20),A) ;
  PUT SKIP(1) EDIT((91)'-')(X(20),A(91)) ;
DO K = 1 TO 3 ;
  PUT SKIP(2) EDIT(CODE3(K), (TAB18(M,K,L) DO L=1 TO 13))
    (X(14),A(6),13 F(7)) ;
END ;
END ;
  PUT PAGE ;
  PUT SKIP(3) EDIT('TABLE #20')(X(5),A(9)) ;

```

```

00037900
00038000
00038100
00038200
00038300
00038400
00038500
00038600
00038700
00038800
00038900
00039000
00039100
00039200
00039300
00039400
00039500
00039600
00039700
00039800
00039900
00040000
00040100
00040200
00040300
00040400
00040500
00040600
00040700
00040800
00040900
00041000
00041100
00041200
00041300
00041400
00041500
00041600
00041700
00041800
00041900
00042000
00042100
00042200
00042300
00042400
00042500
00042600
00042700
00042800
00042900
00043000
00043100
00043200
00043300
00043400
00043500
00043600
00043700
00043800
00043900
00044000
00044100

```

```

DO N = 1 TO 4 ;                                00044200
  PUT SKIP(2) EDIT(CODE1(N))(X(8),A(13)) ;      00044300
  PUT SKIP(2) EDIT(CODE2)(X(20),A) ;            00044400
  PUT SKIP(1) EDIT((91)'-')(X(20),A(91)) ;     00044500
  DO K = 1 TO 3 ;                                00044600
  PUT SKIP(2) EDIT(CODE3(K),(TAB20(N,K,L)DO L=1 TO 13))
    (X(14),A,13 F(7)) ;                          00044700
  END ;                                           00044800
  END ; PUT PAGE ;                               00044900
PUT SKIP(3) EDIT('TABLE #12')(X(5),A(9)) ;      00045000
DO N = 1 TO 4 ;                                00045100
  PUT SKIP(3) ;                                  00045200
  PUT SKIP(2) EDIT(CODE1(N))(X(8),A(13)) ;      00045300
  PUT SKIP(2) EDIT(TIT12)(X(20),A) ;            00045400
  PUT SKIP(1) EDIT((91)'-')(X(20),A(91)) ;     00045500
  DO K = 1 TO 33 ;                               00045600
  PUT SKIP(1) EDIT(CODE12(K),(TAB12(N,K,L)DO L=1 TO 13))
    (X(10),A,13 F(7)) ;                          00045700
  END ; PUT PAGE ;                               00045800
END ; PUT PAGE ;                                00045900
PUT SKIP(3) EDIT('TABLE #13')(X(5),A(9)) ;      00046000
DO N = 1 TO 4 ;                                00046100
  PUT SKIP(2) EDIT(CODE1(N))(X(8),A(13)) ;      00046200
  PUT SKIP(2) EDIT(TIT13)(X(20),A) ;            00046300
  PUT SKIP(1) EDIT((50)'-')(X(20),A(50)) ;     00046400
  DO K = 1 TO 6 ;                               00046500
  PUT SKIP(1) EDIT(CODE13(K),(TAB13(N,K,L)DO L=1 TO 5))
    (X(10),A,5 F(10)) ;                          00046600
  END ;                                           00046700
END ; PUT PAGE ;                               00046800
PUT SKIP(1) EDIT('TABLE #14')(X(5),A(9)) ;      00046900
DO N = 1 TO 4 ;                                00047000
  PUT SKIP(2) EDIT(CODE1(N))(X(8),A(13)) ;      00047100
  PUT SKIP(1) EDIT(TIT14)(X(20),A) ;            00047200
  PUT SKIP(1) EDIT((60)'-')(X(20),A(60)) ;     00047300
  DO K = 1 TO 54 ;                               00047400
  PUT SKIP(1) EDIT(CODE14(K),(TAB14(N,K,L)DO L=1 TO 6))
    (X(14),A,6 F(10)) ;                          00047500
  END ; PUT PAGE ;                               00047600
END ;                                           00047700
PUT SKIP(3) EDIT('TABLE #15')(X(5),A(9)) ;      00047800
DO N = 1 TO 4 ;                                00047900
  PUT SKIP(3) EDIT(CODE1(N))(X(8),A(13)) ;      00048000
  PUT SKIP(2) EDIT('TOTAL')(X(8),A(5)) ;         00048100
  PUT SKIP(2) EDIT(TIT15)(X(20),A) ;            00048200
  PUT SKIP(1) EDIT((91)'-')(X(20),A(91)) ;     00048300
DO K = 1 TO 21 ;                               00048400
  PUT SKIP(2) EDIT(CODE5(K),(TAB15(N,K,L) DO L=1 TO 13))
    (X(13),A,13 F(7)) ;                          00048500
END ; PUT PAGE ;                               00048600
PUT SKIP(2) EDIT('MALE')(X(9),A(4)) ;           00048700
PUT SKIP(2) EDIT(TIT15)(X(20),A) ;             00048800
PUT SKIP(1) EDIT((91)'-')(X(20),A(91)) ;     00048900
DO K = 22 TO 42 ;                              00049000
  PUT SKIP(2) EDIT(CODE5(K),(TAB15(N,K,L) DO L=1 TO 13))
    (X(13),A,13 F(7)) ;                          00049100
END ; PUT PAGE ;                               00049200
PUT SKIP(2) EDIT('FEMALE')(X(7),A(6)) ;        00049300
PUT SKIP(2) EDIT(TIT15)(X(20),A) ;             00049400
PUT SKIP(1) EDIT((91)'-')(X(20),A(91)) ;     00049500
DO K = 43 TO 63 ;                              00049600
  PUT SKIP(2) EDIT(CODE5(K),(TAB15(N,K,L) DO L=1 TO 13))
    (X(13),A,13 F(7)) ;                          00049700
END ; PUT PAGE ;                               00049800
PUT SKIP(2) EDIT('FEMALE')(X(7),A(6)) ;        00049900
PUT SKIP(2) EDIT(TIT15)(X(20),A) ;             00050000
PUT SKIP(1) EDIT((91)'-')(X(20),A(91)) ;     00050100
DO K = 64 TO 84 ;                              00050200
  PUT SKIP(2) EDIT(CODE5(K),(TAB15(N,K,L) DO L=1 TO 13))
    (X(13),A,13 F(7)) ;                          00050300
END ; PUT PAGE ;                               00050400

```

END ;PUT PAGE ;	00050500
END ;	00050600
PUT SKIP(3) EDIT('TABLE #16')(X(5),A(9)) ;	00050700
DO N = 1 TO 4 ;	00050800
PUT SKIP(2) EDIT(CODE1(N))(X(8),A(13)) ;	00050900
PUT SKIP(2) EDIT(TIT16)(X(20),A) ;	00051000
PUT SKIP(1) EDIT((60)'-'')(X(20),A(60)) ;	00051100
DO K = 1 TO 25 ;	00051200
PUT SKIP(2) EDIT(CODE16(K), (TAB16(N,K,L)DO L=1 TO 6))	00051300
(X(14),A,6 F(10)) ;	00051400
END ;PUT PAGE ;	00051500
END ;PUT PAGE ;	00051600
CLOSE FILE(TPI) ;	00051700
BAA : PROC ;	00051800
IF SEX = '1' THEN GO TO A22 ;	00051900
IF SEX ^= '2' THEN GO TO RD ;	00052000
K = 3 ; GO TO A2 ;	00052100
A22 : K = 2 ;	00052200
A2 : DO L = 1 TO 12 ;	00052300
IF BIRM = MONTH(L) THEN GO TO AP1 ;	00052400
END ;	00052500
GO TO T29 ;	00052600
AP1 : B = L+1 ;	00052700
TAB6(N,K,B) = TAB6(N,K,B) + 1 ;	00052800
T29 : DO J = 1 TO 8 ;	00052900
IF MAGE <= Y3(J) THEN GO TO E1 ;	00053000
END ;	00053100
IF MAGE ^= '99' THEN GO TO E11 ;	00053200
C = 11 ;	00053300
E11 : C = 10 ;	00053400
GO TO E111 ;	00053500
E1 : C = J + 1 ;	00053600
E111 : DO M = 1 TO 9 ;	00053700
IF SEQ = KA1(M) THEN GO TO E2 ;	00053800
END ;	00053900
IF SEQ ^= '99' THEN GO TO E22 ;	00054000
D = 12 ;GO TO E222 ;	00054100
E22 : D = 11 ;	00054200
GO TO E222 ;	00054300
E2 : D = M+1 ;	00054400
E222 : TAB8(N,C,D) = TAB8(N,C,D) + 1 ;	00054500
DO L = 1 TO 4 ;	00054600
IF MED <= KA2(L) THEN GO TO AP2 ;	00054700
END ;	00054800
IF MED ^= '99' THEN GO TO AP22 ;	00054900
I = 7 ;GO TO AP333 ;	00055000
AP22 : I = 6 ; GO TO AP333 ;	00055100
AP2 : I = L + 1 ;	00055200
AP333 : TAB9(N,C,I) = TAB9(N,C,I) + 1 ;	00055300
TAB7(N,C,K) = TAB7(N,C,K) + 1 ;	00055400
END BAA ;	00055500
BAB : PROC ;	00055600
IF SSEX = '1' THEN GO TO B33 ;	00055700
K = 3 ; GO TO B3 ;	00055800
B33 : K = 2 ;	00055900
B3 : DO L = 1 TO 12 ;	00056000
IF SBIM = MONTH(L) THEN GO TO B4 ;	00056100
END ;	00056200
GO TO T18 ;	00056300
B4 : B = L + 1 ;	00056400
TAB10(N,K,B) = TAB10(N,K,B) + 1 ;	00056500
T18 : TAB10(N,K,B) = TAB10(N,K,B) + 1 ;	00056600
	00056700

IF SSEX = '1' THEN GO TO P33 ;	00056800
J = 42 ; GO TO C1 ;	00056900
P33: J = 21 ;	00057000
C1: DO M = 1 TO 18 ;	00057100
IF BIY >= YYY(M) THEN GO TO C2 ;	00057200
END ;	00057300
IF BIY = 999 THEN GO TO C22 ;	00057400
R = J+21 ; GO TO C222 ;	00057500
C22: R = J+20 ; GO TO C222 ;	00057600
C2: R = M + J + 1 ;	00057700
C222: DO L = 1 TO 4 ;	00057800
IF SED <= KA2(L) THEN GO TO C3 ;	00057900
END ;	00058000
IF SED = '99' THEN GO TO C33 ;	00058100
I = 7 ; GO TO C333 ;	00058200
C33: I = 6 ; GO TO C333 ;	00058300
C3: I = L + 1 ;	00058400
C333: TAB11(N,R,I) = TAB11(N,R,I) + 1 ;	00058500
DO M = 1 TO 12 ;	00058600
IF SAWO = SS(M) THEN GO TO G2 ;	00058700
END ; GO TO RD1 ;	00058800
G2: E = M + 1 ;	00058900
TAB15(N,R,E) = TAB15(N,R,E) + 1 ;	00059000
RD1: IF SSEX = '1' THEN GO TO SS2 ;	00059100
Q = 22 ; GO TO SS3 ;	00059200
SS2: Q = 11 ;	00059300
SS3: DO M = 4 TO 14 ;	00059400
IF BIY >= YYY(M) THEN GO TO FF2 ;	00059500
END ;	00059600
A = 13 ; GO TO F222 ;	00059700
FF2: A = M - 2 ;	00059800
F222: IF SUBSTR(JKU,1,1) > '0' THEN GO TO F22 ;	00059900
L = 1 ; GO TO F3 ;	00060000
F22: DO L = 2 TO 10 ;	00060100
IF SUBSTR(JKU,1,1) = JIK(L) THEN GO TO F3 ;	00060200
END ; GO TO RD2 ;	00060300
F3: B = L + 1 + Q ;	00060400
TAB12(N,B,A) = TAB12(N,B,A) + 1 ;	00060500
RD2: DO M = 1 TO 4 ;	00060600
IF JIN = JJ(M) THEN GO TO F4 ;	00060700
END ; GO TO RD3 ;	00060800
F4: C = M + 1 ;	00060900
IF SAWO = '11' THEN GO TO F44 ;	00061000
D = 3 ; GO TO F54 ;	00061100
F44: D = 2 ;	00061200
F54: IF SSEX = '1' THEN GO TO F5 ;	00061300
O = 6 ; GO TO F444 ;	00061400
F5: O = 5 ;	00061500
F444: TAB13(N,D,C) = TAB13(N,D,C) + 1 ;	00061600
TAB13(N,O,C) = TAB13(N,O,C) + 1 ;	00061700
RD3: IF SSEX = '1' THEN GO TO Q2 ;	00061800
F = 36 ; GO TO Q22 ;	00061900
Q2: F = 18 ;	00062000
Q22: DO M = 4 TO 18 ;	00062100
IF BIY >= YYY(M) THEN GO TO Q222 ;	00062200
END ;	00062300
IF BIY = 999 THEN GO TO Q3 ;	00062400
S = F + 18 ; GO TO Q33 ;	00062500
Q3: S = F + 17 ; GO TO Q33 ;	00062600
Q222: S = M + F - 2 ;	00062700
Q33: DO L = 1 TO 5 ;	00062800
IF HONS = BBB(L) THEN GO TO R2 ;	00062900
	00063000

```

      END ; GO TO QQSS ;
R2: G = L + 1 ;
      TAB14(N,S,G) = TAB14(N,S,G) + 1 ;
QQSS: IF BIY = 983 THEN GO TO Y6 ;
      IF SSEX = '1' THEN GO TO Y7 ;
      C = 6 ; GO TO Y9 ;
Y7 : C = 5 ; GO TO Y9 ;
Y6 : IF BIY = 984 THEN GO TO RD ;
      IF SSEX = '1' THEN GO TO Y8 ;
      C = 3 ; GO TO Y9 ;
Y8 : C = 2 ;
Y9: BBSS = ((BIM*30)+BID) - ((SBIM*30)+SBID) ;
      IF BBSS < 1 THEN GO TO Y1 ;
      DO M = 1 TO 6 ;
      IF BBSS = YIL(M) THEN GO TO Y2 ;
      END ;
      DO L = 7 TO 10 ;
      IF BBSS <= YIL(L) THEN GO TO Y3 ;
      END ;
      DO K = 11 TO 20 ;
      IF BBSS = YIL(K) THEN GO TO Y4 ;
      END ; GO TO RD ;
Y1 : H = 2 ; GO TO Y5 ;
Y2 : H = M + 3 ; GO TO Y5 ;
Y3 : H = L + 4 ; GO TO Y5 ;
Y4 : H = K + 5 ;
Y5: TAB16(N,H,C) = TAB16(N,H,C) + 1 ;
      END BAB ;
      END LEE ;
/*
//GO.TPI DD UNIT=DISK,DSN=S93.VH.G09;VOL=SER=BOS005,
// DCB=(RECFM=FB,LRECL=100,BLKSIZE=4000),DISP=OLD
//GO.SYSPRINT DD SYSOUT=*
//

```

```

00063100
00063200
00063300
00063400
00063500
00063600
00063700
00063800
00063900
00064000
00064100
00064200
00064300
00064400
00064500
00064600
00064700
00064800
00064900
00065000
00065100
00065200
00065300
00065400
00065500
00065600
00065700
00065800
00065900
00066000
00066100
00066200
00066300
00066400

```

A33,86.055,17:47:52,A33.PROGRAM.LIBRARY(T293)

```
//A33VHWT2 JOB CLASS=A,MSGCLASS=X,NOTIFY=A33,TYPRUN=HOLD          00000100
//TAB EXEC ASMFCLG,REGION=4096K                                   00000200
//ASM.SYSGD DD DSN=&&OBJSET,UNIT=KYYC,SPACE=(80,(5000,50)),      00000300
//          DISP=(MOD,PASS)                                       00000400
//ASM.SYSIN DD *                                                00000500
          PRINT NOGEN                                           00000600
          START 0                                               00000700
ARRY1    DC    2436PL4'0'                                       00000800
ARRY2    DC    2436PL4'0'                                       00000900
ARRY3    DC    2436PL4'0'                                       00001000
ARRY4    DC    2436PL4'0'                                       00001100
ARRY5    DC    2436PL4'0'                                       00001200
ARRY6    DC    2436PL4'0'                                       00001300
ARRYT1   DC    2436PL4'0'                                       00001400
ARRYT2   DC    2436PL4'0'                                       00001500
TAB1     DC    476PL4'0'                                         00001600
TAB2     DC    476PL4'0'                                         00001700
TAB3     DC    476PL4'0'                                         00001800
TAB4     DC    476PL4'0'                                         00001900
TAB5     DC    476PL4'0'                                         00002000
TAB6     DC    476PL4'0'                                         00002100
TABT1    DC    476PL4'0'                                         00002200
TABT2    DC    476PL4'0'                                         00002300
TT1      DC    28PL4'0'                                          00002400
TT2      DC    28PL4'0'                                          00002500
T1       DC    28PL4'0'                                          00002600
T2       DC    28PL4'0'                                          00002700
T3       DC    28PL4'0'                                          00002800
T4       DC    28PL4'0'                                          00002900
T5       DC    28PL4'0'                                          00003000
T6       DC    28PL4'0'                                          00003100
BRANCH1  DC    A(ARRY1,ARRY2,ARRY3,ARRY4,ARRY5,ARRY6)          00003200
BRANCH2  DC    A(TAB1,TAB2,TAB3,TAB4,TAB5,TAB6)                00003300
BRANCH3  DC    A(TT1,TT2,T1,T2,T3,T4,T5,T6)                    00003400
BRANCH4  DC    A(TABT1,TABT2)                                   00003500
BRANCH5  DC    A(ARRYT1,ARRY1,ARRY2,ARRY3)                     00003600
BRANCH6  DC    A(ARRYT2,ARRY4,ARRY5,ARRY6)                     00003700
BRANCH7  DC    A(ARRYT1,ARRYT2,ARRY1,ARRY2,ARRY3,ARRY4,ARRY5,ARRY6) 00003800
BRANCH8  DC    A(TABT1,TABT2,TAB1,TAB2,TAB3,TAB4,TAB5,TAB6)  00003900
START    SAVE    (14,12)                                        00004000
          BALR    2,0                                           00004100
          USING  BOS,2,3                                         00004200
BOS      L      3,=A(BOS+4096)                                   00004300
          B      OPN                                             00004400
          LTORG                                         00004500
OPN      ST     13,SAVE+4                                         00004600
          LA     11,SAVE                                         00004700
          ST     11,8(13)                                         00004800
          LR     13,11                                           00004900
          OPEN   (TPI,(INPUT))                                   00005000
          OPEN   (PR,(OUTPUT))                                   00005100
          MVI    PA,X'89'                                         00005200
          PUT    PR,PA                                           00005300
          GET    TPI,TA                                           00005400
RD       CLI    TA+79,C'2'                                       00005500
SVHH    BE     RD                                               00005600
          CLI    TA+79,C'4'                                       00005700
          BE     RD                                               00005800
          CLI    TA+61,C'8'                                       00005900
          BE     RD                                               00006000
          CLI    TA+66,C'8'                                       00006100
          BE     RD                                               00006200
          CLI    TA+71,C'8'                                       00006300
```



	BE	RD		00006400
	CLI	TA+76,C'8'		00006500
	BE	RD		00006600
	MVZ	TA(80),=80X'F0'		00006700
	CLC	TA+7(3),=C'500'		00006800
	BH	NEXT1		00006900
HKPP	CLC	TA+29(3),=3C'9'		00007000
	BNE	N1		00007100
	L	5,=A(ARRY1+9632)		00007200
	L	6,=A(ARRY4+9632)		00007300
	B	SKIP2		00007400
N1	L	5,=A(ARRY1)		00007500
	L	6,=A(ARRY4)		00007600
	B	NEXT3		00007700
NEXT1	CLI	TA+8,C'6'		00007800
	BH	NEXT2		00007900
	CLC	TA+29(3),=3C'9'		00008000
	BNE	N2		00008100
	L	5,=A(ARRY2+9632)		00008200
	L	6,=A(ARRY5+9632)		00008300
	B	SKIP2		00008400
N2	L	5,=A(ARRY2)		00008500
	L	6,=A(ARRY5)		00008600
	B	NEXT3		00008700
NEXT2	CLC	TA+29(3),=3C'9'		00008800
	BNE	N3		00008900
	L	5,=A(ARRY3+9632)		00009000
	L	6,=A(ARRY6+9632)		00009100
	B	SKIP2		00009200
N3	L	5,=A(ARRY3)		00009300
	L	6,=A(ARRY6)		00009400
NEXT3	PACK	YEAR1(4),TA+29(3)		00009500
	PACK	YEAR2(4),TA+32(2)		00009600
	MP	YEAR1,=P'12'		00009700
	CLC	TA+32(2),=C'99'		00009800
	BE	LVA		00009900
LVA	AP	YEAR1, YEAR2		00010000
	ZAP	YEAR3,=P'984' *** YEAR-CHANGE ***		00010100
	MP	YEAR3,=P'12'		00010200
	AP	YEAR3,=P'6' *** MONTH-CHANGE ***		00010300
	SP	YEAR3, YEAR1		00010400
	ZAP	YEART, YEAR3		00010500
	DP	YEART,=P'12'		00010600
	CP	YEART+4(2),=P'4'		00010700
	BH	HA		00010800
	CLC	TA+36(2),=C'00'		00010900
	BE	LANA		00011000
	MVC	TA+36(2),=C'00'		00011100
	B	LANA		00011200
HA	CP	YEART+4(2),=P'10'		00011300
	BH	HB		00011400
	CLC	TA+36(2),=C'06'		00011500
	BNH	LANA		00011600
	CLC	TA+36(2),=C'99'		00011700
	BE	LANA		00011800
	MVC	TA+36(2),=C'99'		00011900
	B	LANA		00012000
HB	CP	YEART+4(2),=P'13'		00012100
	BH	HC		00012200
	CLC	TA+36(2),=C'09'		00012300
	BNH	LANA		00012400
	CLC	TA+36(2),=C'99'		00012500
	BE	LANA		00012600

	MVC	TA+36(2),=C'99'	00012700
	B	LANA	00012800
HC	CP	YEART+4(2),=P'16'	00012900
	BH	LANA	00013000
	CLC	TA+36(2),=C'12'	00013100
	BNH	LANA	00013200
	CLC	TA+36(2),=C'99'	00013300
	BE	LANA	00013400
	MVC	TA+36(2),=C'99'	00013500
	B	LANA	00013600
LANA	CP	YEART+4(2),=P'15'	00013700
	BH	LANB	00013800
	CLI	TA+38,C'4'	00013900
	BE	LANB	00014000
	CLI	TA+38,C'9'	00014100
	BE	LANB	00014200
	MVI	TA+38,C'4'	00014300
	B	LANB	00014400
LANB	LA	4,NAI	00014500
	LA	9,86	00014600
COMP1	CP	YEART+4(2),0(2,4)	00014700
	BE	SKIP2	00014800
	LA	4,2(4)	00014900
	LA	5,112(5)	00015000
	LA	6,112(6)	00015100
	BCT	9,COMP1	00015200
SKIP2	CLC	TA+36(2),=C'00'	00015300
	BH	SP1	00015400
	A	5,=F'16'	00015500
	B	BRCH1	00015600
SP1	CLC	TA+36(2),=C'06'	00015700
	BH	SP2	00015800
	A	5,=F'32'	00015900
	B	BRCH1	00016000
SP2	CLC	TA+36(2),=C'09'	00016100
	BH	SP3	00016200
	A	5,=F'48'	00016300
	B	BRCH1	00016400
SP3	CLC	TA+36(2),=C'12'	00016500
	BH	SP4	00016600
	A	5,=F'64'	00016700
	B	BRCH1	00016800
SP4	CLC	TA+36(2),=C'99'	00016900
	BNE	SP5	00017000
	A	5,=F'96'	00017100
	B	BRCH1	00017200
SP5	A	5,=F'80'	00017300
BRCH1	CLI	TA+38,C'4'	00017400
	BNE	Q1	00017500
	A	6,=F'16'	00017600
	B	BRCH11	00017700
Q1	CLI	TA+38,C'1'	00017800
	BNE	Q2	00017900
	A	6,=F'32'	00018000
	B	BRCH11	00018100
Q2	CLI	TA+38,C'3'	00018200
	BNE	Q3	00018300
	A	6,=F'48'	00018400
	B	BRCH11	00018500
Q3	CLI	TA+38,C'2'	00018600
	BNE	Q4	00018700
	A	6,=F'64'	00018800
	B	BRCH11	00018900

Q4	CLI	TA+38,C'5'	00019000
	BNE	Q5	00019100
	A	6,=F'80'	00019200
	B	BRCH11	00019300
Q5	A	6,=F'96'	00019400
BRCH11	CLI	TA+28,C'1'	00019500
	BNE	SP6	00019600
	A	5,=F'4'	00019700
	A	6,=F'4'	00019800
	B	BRCH2	00019900
SP6	CLI	TA+28,C'2'	00020000
	BNE	SP7	00020100
	A	5,=F'8'	00020200
	A	6,=F'8'	00020300
	B	BRCH2	00020400
SP7	A	5,=F'12'	00020500
	A	6,=F'12'	00020600
BRCH2	EQU	*	00020700
AP2	AP	0(4,5),=P'1'	00020800
	AP	0(4,6),=P'1'	00020900
BRDD	B	RD	00021000
LAST	L	5,=A(BRANCH1)	00021100
	LA	6,3	00021200
ALPB	L	4,=A(ARRYT1)	00021300
	L	7,0(5)	00021400
	LA	8,87*28	00021500
ALPA	AP	0(4,4),0(4,7)	00021600
	LA	4,4(4)	00021700
	LA	7,4(7)	00021800
	BCT	8,ALPA	00021900
	LA	5,4(5)	00022000
	BCT	6,ALPB	00022100
	L	5,=A(BRANCH1+12)	00022200
	LA	6,3	00022300
ALPD	L	4,=A(ARRYT2)	00022400
	L	7,0(5)	00022500
	LA	8,87*28	00022600
ALPC	AP	0(4,4),0(4,7)	00022700
	LA	4,4(4)	00022800
	LA	7,4(7)	00022900
	BCT	8,ALPC	00023000
	LA	5,4(5)	00023100
	BCT	6,ALPD	00023200
	L	4,=A(BRANCH7)	00023300
	L	5,=A(BRANCH8)	00023400
	LA	6,8	00023500
BLPD	L	7,0(4)	00023600
	L	8,0(5)	00023700
	LA	10,17	00023800
BLPC	LA	11,5	00023900
BLPB	LA	9,28	00024000
BLPA	AP	0(4,8),0(4,7)	00024100
	LA	7,4(7)	00024200
	LA	8,4(8)	00024300
	BCT	9,BLPA	00024400
	S	8,=F'112'	00024500
	BCT	11,BLPB	00024600
	LA	8,112(8)	00024700
	BCT	10,BLPC	00024800
	LA	4,4(4)	00024900
	LA	5,4(5)	00025000
	BCT	6,BLPD	00025100
	L	4,=A(BRANCH3)	00025200

	L	5,=A(BRANCH7)	00025300
	LA	6,8	00025400
CLPC	L	7,0(4)	00025500
	L	8,0(5)	00025600
	LA	10,87	00025700
CLPB	LA	11,28	00025800
CLPA	AP	0(4,7),0(4,8)	00025900
	LA	7,4(7)	00026000
	LA	8,4(8)	00026100
	BCT	11,CLPA	00026200
	S	7,=F'112'	00026300
	BCT	10,CLPB	00026400
	LA	4,4(4)	00026500
	LA	5,4(5)	00026600
	BCT	6,CLPC	00026700
	L	4,=A(BRANCH7)	00026800
	LA	9,8	00026900
DLPC	L	5,0(4)	00027000
	LR	6,5	00027100
	LA	8,87	00027200
DLPB	LA	7,7	00027300
DLPA	AP	0(4,6),4(4,5)	00027400
	AP	0(4,6),8(4,5)	00027500
	AP	0(4,6),12(4,5)	00027600
	LA	6,16(6)	00027700
	LA	5,16(5)	00027800
	BCT	7,DLPA	00027900
	BCT	8,DLPB	00028000
	LA	4,4(4)	00028100
	BCT	9,DLPC	00028200
	L	4,=A(BRANCH8)	00028300
	LA	9,8	00028400
DLPF	L	5,0(4)	00028500
	LR	6,5	00028600
	LA	8,17	00028700
DLPE	LA	7,7	00028800
DLPD	AP	0(4,6),4(4,5)	00028900
	AP	0(4,6),8(4,5)	00029000
	AP	0(4,6),12(4,5)	00029100
	LA	6,16(6)	00029200
	LA	5,16(5)	00029300
	BCT	7,DLPD	00029400
	BCT	8,DLPE	00029500
	LA	4,4(4)	00029600
	BCT	9,DLPF	00029700
	L	4,=A(BRANCH3)	00029800
	LA	9,8	00029900
DLPH	L	5,0(4)	00030000
	LR	6,5	00030100
	LA	7,7	00030200
DLPG	AP	0(4,6),4(4,5)	00030300
	AP	0(4,6),8(4,5)	00030400
	AP	0(4,6),12(4,5)	00030500
	LA	6,16(6)	00030600
	LA	5,16(5)	00030700
	BCT	7,DLPG	00030800
	LA	4,4(4)	00030900
	BCT	9,DLPH	00031000
	L	4,=A(BRANCH7)	00031100
	LA	9,8	00031200
ELPBB	L	5,0(4)	00031300
	LR	6,5	00031400
	LA	8,87	00031500

ELPB	LA	7,6	00031600
ELPA	AP	0(4,6),16(4,5)	00031700
	AP	4(4,6),20(4,5)	00031800
	AP	8(4,6),24(4,5)	00031900
	AP	12(4,6),28(4,5)	00032000
	LA	5,16(5)	00032100
	BCT	7,ELPA	00032200
	A	6,=F'112'	00032300
	A	5,=F'16'	00032400
	BCT	8,ELPB	00032500
	LA	4,4(4)	00032600
	BCT	9,ELPBB	00032700
	L	4,=A(BRANCH8)	00032800
	LA	9,8	00032900
ELPE	L	5,0(4)	00033000
	LR	6,5	00033100
	LA	8,17	00033200
ELPD	LA	7,6	00033300
ELPC	AP	0(4,6),16(4,5)	00033400
	AP	4(4,6),20(4,5)	00033500
	AP	8(4,6),24(4,5)	00033600
	AP	12(4,6),28(4,5)	00033700
	LA	5,16(5)	00033800
	BCT	7,ELPC	00033900
	A	6,=F'112'	00034000
	A	5,=F'16'	00034100
	BCT	8,ELPD	00034200
	LA	4,4(4)	00034300
	BCT	9,ELPE	00034400
	L	4,=A(BRANCH3)	00034500
	LA	9,8	00034600
ELPG	L	5,0(4)	00034700
	LR	6,5	00034800
	LA	7,6	00034900
ELPF	AP	0(4,6),16(4,5)	00035000
	AP	4(4,6),20(4,5)	00035100
	AP	8(4,6),24(4,5)	00035200
	AP	12(4,6),28(4,5)	00035300
	LA	5,16(5)	00035400
	BCT	7,ELPF	00035500
	LA	4,4(4)	00035600
	BCT	9,ELPG	00035700
	LA	11, CODE3	00035800
	BAL	12, HEAD1	00035900
	L	4,=A(ARRYT1)	00036000
	L	5,=A(TABT1)	00036100
	L	10,=A(TT1)	00036200
	MVI	SW,C'0'	00036300
	BAL	12,PRIT	00036400
	BAL	12,HEAD4	00036500
	L	4,=A(ARRYT1+64)	00036600
	L	5,=A(TABT1+64)	00036700
	L	10,=A(TT1+64)	00036800
	MVI	SW,C'2'	00036900
	BAL	12,PRIT	00037000
	LA	11, CODE3+13	00037100
	BAL	12, HEAD1	00037200
	L	4,=A(ARRY1)	00037300
	L	5,=A(TAB1)	00037400
	L	10,=A(T1)	00037500
	MVI	SW,C'0'	00037600
	BAL	12,PRIT	00037700
	BAL	12, HEAD4	00037800

L	4,=A(ARRY1+64)	00037900
L	5,=A(TAB1+64)	00038000
L	10,=A(T1+64)	00038100
MVI	SW,C'2'	00038200
BAL	12,PRIT	00038300
LA	11,CODE3+26	00038400
BAL	12,HEAD1	00038500
L	4,=A(ARRY2)	00038600
L	5,=A(TAB2)	00038700
L	10,=A(T2)	00038800
MVI	SW,C'0'	00038900
BAL	12,PRIT	00039000
BAL	12,HEAD4	00039100
L	4,=A(ARRY2+64)	00039200
L	5,=A(TAB2+64)	00039300
L	10,=A(T2+64)	00039400
MVI	SW,C'2'	00039500
BAL	12,PRIT	00039600
LA	11,CODE3+39	00039700
BAL	12,HEAD1	00039800
L	4,=A(ARRY3)	00039900
L	5,=A(TAB3)	00040000
L	10,=A(T3)	00040100
MVI	SW,C'0'	00040200
BAL	12,PRIT	00040300
BAL	12,HEAD4	00040400
L	4,=A(ARRY3+64)	00040500
L	5,=A(TAB3+64)	00040600
L	10,=A(T3+64)	00040700
MVI	SW,C'2'	00040800
BAL	12,PRIT	00040900
LA	11,CODE3	00041000
BAL	12,HEAD2	00041100
L	4,=A(ARRYT2)	00041200
L	5,=A(TABT2)	00041300
L	10,=A(TT2)	00041400
MVI	SW,C'0'	00041500
BAL	12,PRIT	00041600
BAL	12,HEAD5	00041700
L	4,=A(ARRYT2+64)	00041800
L	5,=A(TABT2+64)	00041900
L	10,=A(TT2+64)	00042000
MVI	SW,C'2'	00042100
BAL	12,PRIT	00042200
LA	11,CODE3+13	00042300
BAL	12,HEAD2	00042400
L	4,=A(ARRY4)	00042500
L	5,=A(TAB4)	00042600
L	10,=A(T4)	00042700
MVI	SW,C'0'	00042800
BAL	12,PRIT	00042900
BAL	12,HEAD5	00043000
L	4,=A(ARRY4+64)	00043100
L	5,=A(TAB4+64)	00043200
L	10,=A(T4+64)	00043300
MVI	SW,C'2'	00043400
BAL	12,PRIT	00043500
LA	11,CODE3+26	00043600
BAL	12,HEAD2	00043700
L	4,=A(ARRY5)	00043800
L	5,=A(TAB5)	00043900
L	10,=A(T5)	00044000
MVI	SW,C'0'	00044100

	BAL	12,PRIT	00044200
	BAL	12,HEAD5	00044300
	L	4,=A(ARRY5+64)	00044400
	L	5,=A(TAB5+64)	00044500
	L	10,=A(T5+64)	00044600
	MVI	SW,C'2'	00044700
	BAL	12,PRIT	00044800
	LA	11,CODE3+39	00044900
	BAL	12,HEAD2	00045000
	L	4,=A(ARRY6)	00045100
	L	5,=A(TAB6)	00045200
	L	10,=A(T6)	00045300
	MVI	SW,C'0'	00045400
	BAL	12,PRIT	00045500
	BAL	12,HEAD5	00045600
	L	4,=A(ARRY6+64)	00045700
	L	5,=A(TAB6+64)	00045800
	L	10,=A(T6+64)	00045900
	MVI	SW,C'2'	00046000
	BAL	12,PRIT	00046100
PRIT	B	CLOSE	00046200
	LA	11,17	00046300
	LA	8,CODE1	00046400
	LA	9,CODE2	00046500
	CLI	SW,C'0'	00046600
	BE	NA	00046700
	LA	6,12	00046800
	B	NB	00046900
NA	LA	6,16	00047000
NB	LA	7,PA	00047100
	MVC	PA+3(5),=C'TOTAL'	00047200
ANNA	MVC	10(6,7),=X'402020202120'	00047300
	ED	10(6,7),1(10)	00047400
	LA	7,7(7)	00047500
	LA	10,4(10)	00047600
	BCT	6,ANNA	00047700
	MVI	PA,X'11'	00047800
	PUT	PR,PA	00047900
	MVC	PA,PA-1	00048000
PT4	CLI	SW,C'0'	00048100
	BE	XA	00048200
	LA	6,12	00048300
	B	XB	00048400
XA	LA	6,16	00048500
XB	LA	7,PA+3	00048600
	MVC	0(5,7),0(8)	00048700
TR1	MVC	10(6,7),=X'402020202120'	00048800
	ED	10(6,7),1(5)	00048900
	LA	7,7(7)	00049000
	LA	5,4(5)	00049100
	BCT	6,TR1	00049200
	MVI	PA,X'11'	00049300
	PUT	PR,PA	00049400
	MVC	PA,PA-1	00049500
	CLI	SW,C'0'	00049600
	BE	TAD1	00049700
	A	5,=F'64'	00049800
	B	TAD	00049900
TAD1	A	5,=F'48'	00050000
TAD	LA	10,5	00050100
PT3	CLI	SW,C'0'	00050200
	BE	XC	00050300
	LA	6,12	00050400

	B	XD	00050500
XC	LA	6,16	00050600
XD	LA	7,PA+3	00050700
	MVC	3(2,7),0(9)	00050800
PT2	MVC	10(6,7),=X'402020202120'	00050900
	ED	10(6,7),1(4)	00051000
	LA	7,7(7)	00051100
	LA	4,4(4)	00051200
	BCT	6,PT2	00051300
	MVI	PA,X'09'	00051400
	PUT	PR,PA	00051500
	MVC	PA,PA-1	00051600
	CLI	SW,C'0'	00051700
	BE	XE	00051800
	A	4,=F'64'	00051900
	B	XF	00052000
XE	A	4,=F'48'	00052100
XF	LA	9,2(9)	00052200
	BCT	10,PT3	00052300
	MVI	PA,X'11'	00052400
	PUT	PR,PA	00052500
	MVC	PA,PA-1	00052600
	LA	8,5(8)	00052700
	BCT	11,PT4	00052800
	LA	10,2	00052900
PT6	CLI	SW,C'0'	00053000
	BE	XI	00053100
	LA	6,12	00053200
	B	XJ	00053300
XI	LA	6,16	00053400
XJ	LA	7,PA+3	00053500
	MVC	3(2,7),0(9)	00053600
PT5	MVC	10(6,7),=X'402020202120'	00053700
	ED	10(6,7),1(4)	00053800
	LA	7,7(7)	00053900
	LA	4,4(4)	00054000
	BCT	6,PT5	00054100
	MVI	PA,X'11'	00054200
	PUT	PR,PA	00054300
	MVC	PA,PA-1	00054400
	CLI	SW,C'0'	00054500
	BE	XK	00054600
	A	4,=F'64'	00054700
	B	XL	00054800
XK	A	4,=F'48'	00054900
XL	LA	9,2(9)	00055000
	BCT	10,PT6	00055100
	BR	12	00055200
HEAD1	MVI	PA,X'09'	00055300
	PUT	PR,PA	00055400
	MVC	PA,PA-1	00055500
	MVC	PA+5(13),0(11)	00055600
	MVC	PA+40(10),=C'TABLE #3-1'	00055700
	MVI	PA,X'19'	00055800
	PUT	PR,PA	00055900
	MVC	PA,PA-1	00056000
	MVC	PA(112),=C' AGE	TOTAL
		VER ATTRNDING	PRIMARY SCHOOL
		DDLE SCHOOL'	MI*00056200
	MVI	PA,X'09'	00056300
	PUT	PR,PA	00056400
	MVC	PA,PA-1	00056500
	MVC	PA(123),=123C'-'	00056600
			00056700



	MVI	PA,X'11'										00056800
	PUT	PR,PA										00056900
	MVC	PA,PA-1										00057000
	MVC	PA+11(115),=C'TOTAL	MALE	FEMALE	UNKNOW	TOTAL	MALE	F	*	00057100		
		EMALE UNKNOW	TOTAL	MALE	FEMALE	UNKNOW	TOTAL	MALE	F	*	00057200	
		EMALE UNKNOW										00057300
	MVI	PA,X'11'										00057400
	PUT	PR,PA										00057500
	MVC	PA,PA-1										00057600
	BR	12										00057700
HEAD2	MVI	PA,X'89'										00057800
	PUT	PR,PA										00057900
	MVC	PA+5(13),0(11)										00058000
	MVC	PA+44(10),=C'TABLE #2-1'										00058100
	MVI	PA,X'19'										00058200
	PUT	PR,PA										00058300
	MVC	PA,PA-1										00058400
	MVC	PA(110),=C' AGE				TOTAL						*00058500
		SINGLE				MARIED						*00058600
		DIVORCED'										00058700
	MVI	PA,X'09'										00058800
	PUT	PR,PA										00058900
	MVC	PA,PA-1										00059000
	MVC	PA(121),=121C'-'										00059100
	MVI	PA,X'11'										00059200
	PUT	PR,PA										00059300
	MVC	PA,PA-1										00059400
	MVC	PA+11(115),=C'TOTAL	MALE	FEMALE	UNKNOW	TOTAL	MALE	F	*	00059500		
		EMALE UNKNOW	TOTAL	MALE	FEMALE	UNKNOW	TOTAL	MALE	F	*	00059600	
		EMALE UNKNOW										00059700
	MVI	PA,X'11'										00059800
	PUT	PR,PA										00059900
	MVC	PA,PA-1										00060000
	BR	12										00060100
HEAD4	MVI	PA,X'89'										00060200
	PUT	PR,PA										00060300
	MVC	PA+5(13),0(11)										00060400
	MVC	PA+30(10),=C'TABLE #3-2'										00060500
	MVI	PA,X'19'										00060600
	PUT	PR,PA										00060700
	MVC	PA,PA-1										00060800
	MVC	PA+20(61),=C'HIGH SCHOOL				COLLEGE OR HIGHER						*00060900
		UNKNOW'										00061000
	MVI	PA,X'09'										00061100
	PUT	PR,PA										00061200
	MVC	PA,PA-1										00061300
	MVC	PA+4(89),=89C'-'										00061400
	MVI	PA,X'11'										00061500
	PUT	PR,PA										00061600
	MVC	PA,PA-1										00061700
	MVC	PA+11(82),=C'TOTAL	MALE	FEMALE	UNKNOW	TOTAL	MALE	FE	*	00061800		
		MALE UNKNOW	TOTAL	MALE	FEMALE	UNKNOW'						00061900
		MALE UNKNOW										00062000
	MVI	PA,X'11'										00062100
	PUT	PR,PA										00062200
	MVC	PA,PA-1										00062300
	BR	12										00062400
HEAD5	MVI	PA,X'89'										00062500
	PUT	PR,PA										00062600
	MVC	PA+5(13),0(11)										00062700
	MVC	PA+30(10),=C'TABLE #2-2'										00062800
	MVI	PA,X'19'										00062900
	PUT	PR,PA										00063000
	MVC	PA,PA-1										00063000

```

MVC PA+19(63),=C'WIDOWED          APART          *00063100
      UNKNOW'                          00063200
MVI PA,X'09'                          00063300
PUT PR,PA                              00063400
MVC PA,PA-1                            00063500
MVC PA(89),=89C'-'                    00063600
MVI PA,X'11'                          00063700
PUT PR,PA                              00063800
MVC PA,PA-1                            00063900
MVC PA+11(82),=C'TOTAL MALE FEMALE UNKNOW TOTAL MALE FE*00064000
      MALE UNKNOW TOTAL MALE FEMALE UNKNOW' 00064100
MVI PA,X'11'                          00064200
PUT PR,PA                              00064300
MVC PA,PA-1                            00064400
BR 12                                  00064500
CLOSE CLOSE (TPI)                      00064600
      CLOSE (PR)                        00064700
L 13,SAVE+4                            00064800
RETURN (14,12)                          00064900
TPI DCB DDNAME=TPI,DSORG=PS,BLKSIZE=20000,LRECL=100,BUFNO=2 *00065000
      RECFM=FB,EODAD=LAST,MACRF=GM      00065100
PR DCB DDNAME=PR,BUFNO=1,BLKSIZE=133,RECFM=FM,DSORG=PS, *00065200
      MACRF=PM                          00065300
SAVE DS 18F                            00065400
TA DS CL80                              00065500
C DC X'40'                              00065600
PA DS CL133                             00065700
YEAR DC PL3'0'                          00065800
SW DC C'0'                              00065900
SAVE1 DS F                              00066000
YEAR2 DC PL8'0'                         00066100
YEAR1 DC PL4'0'                         00066200
YEAR2 DC PL4'0'                         00066300
YEAR3 DC PL4'0'                         00066400
YEAR4 DC PL4'0'                         00066500
NAI DC PL2'00,01,02,03,04,05,06,07,08,09,10,11,12,13,14,15,16' 00066600
      DC PL2'17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33' 00066700
      DC PL2'34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50' 00066800
      DC PL2'51,52,53,54,55,56,57,58,59,60,61,62,63,64,65,66,67' 00066900
      DC PL2'68,69,70,71,72,73,74,75,76,77,78,79,80,81,82,83,84' 00067000
      DC PL2'85'                          00067100
CODE1 DC C' 0-4 5-910-1415-1920-2425-2930-3435-3940-4445-49' 00067200
      DC C'50-5455-5960-6465-6970-7475-7980-8485+ MISAG' 00067300
CODE2 DC C' 0 1 2 3 4 5 6 7 8 91011121314151617181920212223242526' 00067400
      DC C'2728293031323334353637383940414243444546474849505152' 00067500
      DC C'5354555657585960616263646566676869707172737475767778' 00067600
      DC C'79808182838485UN'              00067700
CODE3 DC C'WHOLE COUNTRYGREAT CITY SI : BU GUN : BU 00067800
AJO DC C'015024046077109231233314442' 00067900
      END START                          00068000
/*                                       00068100
//GO.TPI DD UNIT=DISK,DISP=(OLD,KEEP),DSN=A33.VH.MS, 00068200
// DCB=(BLKSIZE=20000,LRECL=100,RECFM=FB),VOL=SER=BOS005 00068300
//GO.SYSPRINT DD SYSOUT=* 00068400
//GO.PR DD SYSOUT=* 00068500
//                                       00068600

```

```

//A33VHMAP JOB CLASS=A,MSGCLASS=X,NOTIFY=A33                00000100
//STEP1 EXEC DFDMAPH,LMBR=KKHMAP                            00000200
//CARD DD *                                                  00000300
PRINT NOGEN                                                00000400
KKHMAP DFHMSD TYPE=&SYSPARM,MODE=INOUT,CTRL=(FREEKB,FRSET), *00000500
        LANG=PLI,TIOAPFX=YES,STORAGE=AUTO,EXTATT=YES,PS=X'00' 00000600
MENU DFHMDI SIZE=(24,80),LINE=1,COLUMN=1                  00000700
DFHMDF POS=(3,14),LENGTH=78,ATTRB=(BRT,PROT),PS=X'F8', *00000800
*HG/36 INITIAL='RK RN RHKS FL WJS TKS GHK                00000900
DFHMDF POS=(7,14),LENGTH=78,ATTRB=PROT,PS=X'F8', *00001000
*HG/36 INITIAL='DKFOQJSGHWND XORDLFGKDU "KEY IN"         00001100
DFHMDF POS=(9,14),LENGTH=78,ATTRB=PROT,PS=X'F8', *00001200
*HG/36 INITIAL='1---> RKRN GHKRDLSVUS("SEARCH")         00001300
DFHMDF POS=(11,14),LENGTH=78,ATTRB=PROT,PS=X'F8', *00001400
*HG/36 INITIAL='2---> RKRN TNWJDVUS("UPDATE")            00001500
DFHMDF POS=(13,14),LENGTH=78,ATTRB=PROT,PS=X'F8', *00001600
*HG/36 INITIAL='3---> RKRN TKQDLQVUS("INSERT")          00001700
DFHMDF POS=(15,14),LENGTH=78,ATTRB=PROT,PS=X'F8', *00001800
*HG/36 INITIAL='4---> RKRN TKRWPVUS("DELETE")           00001900
DFHMDF POS=(17,14),LENGTH=78,ATTRB=PROT,PS=X'F8', *00002000
*HG/36 INITIAL='5---> RKRN DLEHDVUS("MOVE")             00002100
DFHMDF POS=(19,14),LENGTH=78,ATTRB=PROT,PS=X'F8', *00002200
*HG/36 INITIAL='6---> VMFHRMFOA QMX("END")              00002300
DFHMDF POS=(22,10),LENGTH=4,ATTRB=BRT,INITIAL='----'    00002400
INFH DFHMDI POS=(22,16),LENGTH=2,ATTRB=(IC,FSET)        00002500
SCRE DFHMDI SIZE=(24,80),LINE=1,COLUMN=1                00002600
DFHMDF POS=(1,2),LENGTH=78,ATTRB=(BRT,PROT),PS=X'F8', *00002700
*HG/44 INITIAL='-----RK RN TK GKD-----'              00002800
DFHMDF POS=(3,10),LENGTH=68,ATTRB=(BRT,PROT), *00002900
        INITIAL='1 2-4 6 7 8 9 22 5' 00003000
DFHMDF POS=(4,2),LENGTH=78,ATTRB=(BRT,PROT),PS=X'F8', *00003100
*HG/54 INITIAL='DLFFUS WHTK RK RN RJTHRNK QNWJR GOD WJD WN 00003200
        TH' 00003300
DFHMDF POS=(5,2),LENGTH=78,ATTRB=(BRT,PROT),PS=X'F8', *00003400
*HG/45 INITIAL='QJSGH QJSGH QJS GH CJDBDURRN RUR RN DUR 00003500
ILNO DFHMDI POS=(7,2),LENGTH=5,ATTRB=(IC,FSET,UNPROT) 00003600
DFHMDF POS=(7,8),LENGTH=1,ATTRB=(PROT,ASKIP) 00003700
JONO DFHMDI POS=(7,10),LENGTH=4,ATTRB=(FSET,UNPROT) 00003800
GANO DFHMDI POS=(7,16),LENGTH=8,ATTRB=(FSET,UNPROT) 00003900
GEO DFHMDI POS=(7,26),LENGTH=4,ATTRB=(FSET,UNPROT) 00004000
DFHMDF POS=(7,31),LENGTH=4,ATTRB=(PROT,ASKIP) 00004100
BUJUK DFHMDI POS=(7,36),LENGTH=2,ATTRB=(FSET,UNPROT) 00004200
DFHMDF POS=(7,39),LENGTH=2,ATTRB=(PROT,ASKIP) 00004300
HANG DFHMDI POS=(7,42),LENGTH=6,ATTRB=(UNPROT,FSET) 00004400
DFHMDF POS=(7,49),LENGTH=2,ATTRB=(PROT,ASKIP) 00004500
JUSO DFHMDI POS=(7,52),LENGTH=28,ATTRB=(UNPROT,FSET),PS=X'F8' 00004600
DFHMDF POS=(9,2),LENGTH=78,ATTRB=(BRT,PROT),PS=X'F8', *00004700
*HG/44 INITIAL='-----RK RN DNJS TK GKD-----'        00004800
DFHMDF POS=(11,2),LENGTH=78,ATTRB=(BRT,PROT), *00004900
        INITIAL='10 11 12 13 14 15 16 17 00005000
        18 19 20 21' 00005100
DFHMDF POS=(12,2),LENGTH=78,ATTRB=(BRT,PROT),PS=X'F8', *00005200
*HG/58 INITIAL='QJS TOD SUS GO CNF DLATLS WJS' 00005300
        DLQ QL RYCP' 00005400
DFHMDF POS=(13,2),LENGTH=78,ATTRB=(BRT,PROT),PS=X'F8', *00005500
*HG/71 INITIAL='GH TJD AUD RHKSRA TJD DNJF DLF GKRFUR DN CHGH' 00005600
        S TKS DUQN DLFWK RH 4 ' 00005700
NUM1 DFHMDI POS=(15,2),LENGTH=2,ATTRB=(FSET,UNPROT) 00005800
DFHMDF POS=(15,5),LENGTH=2,ATTRB=(PROT,ASKIP) 00005900
NAME1 DFHMDI POS=(15,8),LENGTH=6,ATTRB=(FSET,UNPROT),PS=X'F8' 00006000
DFHMDF POS=(15,15),LENGTH=2,ATTRB=(PROT,ASKIP) 00006100
REL1 DFHMDI POS=(15,18),LENGTH=4,ATTRB=(FSET,UNPROT) 00006200
SEX1 DFHMDI POS=(15,24),LENGTH=1,ATTRB=(FSET,UNPROT) 00006300

```

	DFHMDF POS=(15,26),LENGTH=1,ATTRB=(PROT,ASKIP)	00006400
BIR1	DFHMDF POS=(15,28),LENGTH=7,ATTRB=(FSET,UNPROT)	00006500
HAK1	DFHMDF POS=(15,37),LENGTH=3,ATTRB=(FSET,UNPROT)	00006600
BAE1	DFHMDF POS=(15,42),LENGTH=1,ATTRB=(FSET,UNPROT)	00006700
CHO1	DFHMDF POS=(15,45),LENGTH=4,ATTRB=(FSET,UNPROT)	00006800
	DFHMDF POS=(15,50),LENGTH=1,ATTRB=(PROT,ASKIP)	00006900
CHL1	DFHMDF POS=(15,52),LENGTH=2,ATTRB=(FSET,UNPROT)	00007000
IMSN1	DFHMDF POS=(15,56),LENGTH=5,ATTRB=(FSET,UNPROT)	00007100
JUN1	DFHMDF POS=(15,63),LENGTH=6,ATTRB=(FSET,UNPROT)	00007200
BIG01	DFHMDF POS=(15,71),LENGTH=1,ATTRB=(FSET,UNPROT)	00007300
CHGA1	DFHMDF POS=(15,74),LENGTH=5,ATTRB=(FSET,UNPROT)	00007400
NUM2	DFHMDF POS=(16,2),LENGTH=2,ATTRB=(FSET,UNPROT)	00007500
	DFHMDF POS=(16,5),LENGTH=2,ATTRB=(PROT,ASKIP)	00007600
NAME2	DFHMDF POS=(16,8),LENGTH=6,ATTRB=(FSET,UNPROT),PS=X'F8'	00007700
	DFHMDF POS=(16,15),LENGTH=6,ATTRB=(PROT,ASKIP)	00007800
REL2	DFHMDF POS=(16,18),LENGTH=4,ATTRB=(FSET,UNPROT)	00007900
SEX2	DFHMDF POS=(16,24),LENGTH=1,ATTRB=(FSET,UNPROT)	00008000
	DFHMDF POS=(16,26),LENGTH=1,ATTRB=(PROT,ASKIP)	00008100
BIR2	DFHMDF POS=(16,28),LENGTH=7,ATTRB=(FSET,UNPROT)	00008200
HAK2	DFHMDF POS=(16,37),LENGTH=3,ATTRB=(FSET,UNPROT)	00008300
BAE2	DFHMDF POS=(16,42),LENGTH=1,ATTRB=(FSET,UNPROT)	00008400
CHO2	DFHMDF POS=(16,45),LENGTH=4,ATTRB=(FSET,UNPROT)	00008500
	DFHMDF POS=(16,50),LENGTH=1,ATTRB=(PROT,ASKIP)	00008600
CHL2	DFHMDF POS=(16,52),LENGTH=2,ATTRB=(FSET,UNPROT)	00008700
IMSN2	DFHMDF POS=(16,56),LENGTH=5,ATTRB=(FSET,UNPROT)	00008800
JUN2	DFHMDF POS=(16,63),LENGTH=6,ATTRB=(FSET,UNPROT)	00008900
BIG02	DFHMDF POS=(16,71),LENGTH=1,ATTRB=(FSET,UNPROT)	00009000
CHGA2	DFHMDF POS=(16,74),LENGTH=5,ATTRB=(FSET,UNPROT)	00009100
NUM3	DFHMDF POS=(17,2),LENGTH=2,ATTRB=(FSET,UNPROT)	00009200
	DFHMDF POS=(17,5),LENGTH=2,ATTRB=(PROT,ASKIP)	00009300
NAME3	DFHMDF POS=(17,8),LENGTH=6,ATTRB=(FSET,UNPROT),PS=X'F8'	00009400
	DFHMDF POS=(17,15),LENGTH=6,ATTRB=(PROT,ASKIP)	00009500
REL3	DFHMDF POS=(17,18),LENGTH=4,ATTRB=(FSET,UNPROT)	00009600
SEX3	DFHMDF POS=(17,24),LENGTH=1,ATTRB=(FSET,UNPROT)	00009700
	DFHMDF POS=(17,26),LENGTH=1,ATTRB=(PROT,ASKIP)	00009800
BIR3	DFHMDF POS=(17,28),LENGTH=7,ATTRB=(FSET,UNPROT)	00009900
HAK3	DFHMDF POS=(17,37),LENGTH=3,ATTRB=(FSET,UNPROT)	00010000
BAE3	DFHMDF POS=(17,42),LENGTH=1,ATTRB=(FSET,UNPROT)	00010100
CHO3	DFHMDF POS=(17,45),LENGTH=4,ATTRB=(FSET,UNPROT)	00010200
	DFHMDF POS=(17,50),LENGTH=1,ATTRB=(PROT,ASKIP)	00010300
CHL3	DFHMDF POS=(17,52),LENGTH=2,ATTRB=(FSET,UNPROT)	00010400
IMSN3	DFHMDF POS=(17,56),LENGTH=5,ATTRB=(FSET,UNPROT)	00010500
JUN3	DFHMDF POS=(17,63),LENGTH=6,ATTRB=(FSET,UNPROT)	00010600
BIG03	DFHMDF POS=(17,71),LENGTH=1,ATTRB=(FSET,UNPROT)	00010700
CHGA3	DFHMDF POS=(17,74),LENGTH=5,ATTRB=(FSET,UNPROT)	00010800
NUM4	DFHMDF POS=(18,2),LENGTH=2,ATTRB=(FSET,UNPROT)	00010900
	DFHMDF POS=(18,5),LENGTH=2,ATTRB=(PROT,ASKIP)	00011000
NAME4	DFHMDF POS=(18,8),LENGTH=6,ATTRB=(FSET,UNPROT),PS=X'F8'	00011100
	DFHMDF POS=(18,15),LENGTH=6,ATTRB=(PROT,ASKIP)	00011200
REL4	DFHMDF POS=(18,18),LENGTH=4,ATTRB=(FSET,UNPROT)	00011300
SEX4	DFHMDF POS=(18,24),LENGTH=1,ATTRB=(FSET,UNPROT)	00011400
	DFHMDF POS=(18,26),LENGTH=1,ATTRB=(PROT,ASKIP)	00011500
BIR4	DFHMDF POS=(18,28),LENGTH=7,ATTRB=(FSET,UNPROT)	00011600
HAK4	DFHMDF POS=(18,37),LENGTH=3,ATTRB=(FSET,UNPROT)	00011700
BAE4	DFHMDF POS=(18,42),LENGTH=1,ATTRB=(FSET,UNPROT)	00011800
CHO4	DFHMDF POS=(18,45),LENGTH=4,ATTRB=(FSET,UNPROT)	00011900
	DFHMDF POS=(18,50),LENGTH=1,ATTRB=(PROT,ASKIP)	00012000
CHL4	DFHMDF POS=(18,52),LENGTH=2,ATTRB=(FSET,UNPROT)	00012100
IMSN4	DFHMDF POS=(18,56),LENGTH=5,ATTRB=(FSET,UNPROT)	00012200
JUN4	DFHMDF POS=(18,63),LENGTH=6,ATTRB=(FSET,UNPROT)	00012300
BIG04	DFHMDF POS=(18,71),LENGTH=1,ATTRB=(FSET,UNPROT)	00012400
CHGA4	DFHMDF POS=(18,74),LENGTH=5,ATTRB=(FSET,UNPROT)	00012500
NUM5	DFHMDF POS=(19,2),LENGTH=2,ATTRB=(FSET,UNPROT)	00012600

	DFHMDF	POS=(19,5),LENGTH=2,ATTRB=(PROT,ASKIP)	00012700
NAME5	DFHMDF	POS=(19,8),LENGTH=6,ATTRB=(FSET,UNPROT),PS=X'F8'	00012800
	DFHMDF	POS=(19,15),LENGTH=2,ATTRB=(PROT,ASKIP)	00012900
REL5	DFHMDF	POS=(19,18),LENGTH=4,ATTRB=(FSET,UNPROT)	00013000
SEX5	DFHMDF	POS=(19,24),LENGTH=1,ATTRB=(FSET,UNPROT)	00013100
	DFHMDF	POS=(19,26),LENGTH=1,ATTRB=(PROT,ASKIP)	00013200
BIR5	DFHMDF	POS=(19,28),LENGTH=7,ATTRB=(FSET,UNPROT)	00013300
HAK5	DFHMDF	POS=(19,37),LENGTH=3,ATTRB=(FSET,UNPROT)	00013400
BAE5	DFHMDF	POS=(19,42),LENGTH=1,ATTRB=(FSET,UNPROT)	00013500
CH05	DFHMDF	POS=(19,45),LENGTH=4,ATTRB=(FSET,UNPROT)	00013600
	DFHMDF	POS=(19,50),LENGTH=1,ATTRB=(PROT,ASKIP)	00013700
CHL5	DFHMDF	POS=(19,52),LENGTH=2,ATTRB=(FSET,UNPROT)	00013800
IMSN5	DFHMDF	POS=(19,56),LENGTH=5,ATTRB=(FSET,UNPROT)	00013900
JUN5	DFHMDF	POS=(19,63),LENGTH=6,ATTRB=(FSET,UNPROT)	00014000
BIG05	DFHMDF	POS=(19,71),LENGTH=1,ATTRB=(FSET,UNPROT)	00014100
CHGA5	DFHMDF	POS=(19,74),LENGTH=5,ATTRB=(FSET,UNPROT)	00014200
NUM6	DFHMDF	POS=(20,2),LENGTH=2,ATTRB=(FSET,UNPROT)	00014300
	DFHMDF	POS=(20,5),LENGTH=2,ATTRB=(PROT,ASKIP)	00014400
NAME6	DFHMDF	POS=(20,8),LENGTH=6,ATTRB=(FSET,UNPROT),PS=X'F8'	00014500
	DFHMDF	POS=(20,15),LENGTH=2,ATTRB=(PROT,ASKIP)	00014600
REL6	DFHMDF	POS=(20,18),LENGTH=4,ATTRB=(FSET,UNPROT)	00014700
SEX6	DFHMDF	POS=(20,24),LENGTH=1,ATTRB=(FSET,UNPROT)	00014800
	DFHMDF	POS=(20,26),LENGTH=1,ATTRB=(PROT,ASKIP)	00014900
BIR6	DFHMDF	POS=(20,28),LENGTH=7,ATTRB=(FSET,UNPROT)	00015000
HAK6	DFHMDF	POS=(20,37),LENGTH=3,ATTRB=(FSET,UNPROT)	00015100
BAE6	DFHMDF	POS=(20,42),LENGTH=1,ATTRB=(FSET,UNPROT)	00015200
CH06	DFHMDF	POS=(20,45),LENGTH=4,ATTRB=(FSET,UNPROT)	00015300
	DFHMDF	POS=(20,50),LENGTH=1,ATTRB=(PROT,ASKIP)	00015400
CHL6	DFHMDF	POS=(20,52),LENGTH=2,ATTRB=(FSET,UNPROT)	00015500
IMSN6	DFHMDF	POS=(20,56),LENGTH=5,ATTRB=(FSET,UNPROT)	00015600
JUN6	DFHMDF	POS=(20,63),LENGTH=6,ATTRB=(FSET,UNPROT)	00015700
BIG06	DFHMDF	POS=(20,71),LENGTH=1,ATTRB=(FSET,UNPROT)	00015800
CHGA6	DFHMDF	POS=(20,74),LENGTH=5,ATTRB=(FSET,UNPROT)	00015900
NUM7	DFHMDF	POS=(21,2),LENGTH=2,ATTRB=(FSET,UNPROT)	00016000
	DFHMDF	POS=(21,5),LENGTH=2,ATTRB=(PROT,ASKIP)	00016100
NAME7	DFHMDF	POS=(21,8),LENGTH=6,ATTRB=(FSET,UNPROT),PS=X'F8'	00016200
	DFHMDF	POS=(21,15),LENGTH=2,ATTRB=(PROT,ASKIP)	00016300
REL7	DFHMDF	POS=(21,18),LENGTH=4,ATTRB=(FSET,UNPROT)	00016400
SEX7	DFHMDF	POS=(21,24),LENGTH=1,ATTRB=(FSET,UNPROT)	00016500
	DFHMDF	POS=(21,26),LENGTH=1,ATTRB=(PROT,ASKIP)	00016600
BIR7	DFHMDF	POS=(21,28),LENGTH=7,ATTRB=(FSET,UNPROT)	00016700
HAK7	DFHMDF	POS=(21,37),LENGTH=3,ATTRB=(FSET,UNPROT)	00016800
BAE7	DFHMDF	POS=(21,42),LENGTH=1,ATTRB=(FSET,UNPROT)	00016900
CH07	DFHMDF	POS=(21,45),LENGTH=4,ATTRB=(FSET,UNPROT)	00017000
	DFHMDF	POS=(21,50),LENGTH=1,ATTRB=(PROT,ASKIP)	00017100
CHL7	DFHMDF	POS=(21,52),LENGTH=2,ATTRB=(FSET,UNPROT)	00017200
IMSN7	DFHMDF	POS=(21,56),LENGTH=5,ATTRB=(FSET,UNPROT)	00017300
JUN7	DFHMDF	POS=(21,63),LENGTH=6,ATTRB=(FSET,UNPROT)	00017400
BIG07	DFHMDF	POS=(21,71),LENGTH=1,ATTRB=(FSET,UNPROT)	00017500
CHGA7	DFHMDF	POS=(21,74),LENGTH=5,ATTRB=(FSET,UNPROT)	00017600
	DFHMDF	POS=(23,61),LENGTH=4,ATTRB=BRT,INITIAL='===>'	00017700
INCR	DFHMDF	POS=(23,67),LENGTH=1,ATTRB=(IC,FSET),INITIAL='F'	00017800
*			00017900
INST	DFHMDF	SIZE=(24,80),LINE=1,COLUMN=1	00018000
	DFHMDF	POS=(4,10),LENGTH=78,ATTRB=(BRT,PROT),PS=X'F8',	*00018100
*HG/22		INITIAL='RK RN TKR DLQ VUS'	00018200
	DFHMDF	POS=(20,10),LENGTH=4,ATTRB=(PROT,BRT),INITIAL='---->'	00018300
ADD	DFHMDF	POS=(20,16),LENGTH=12,ATTRB=(IC,FSET),	*00018400
		INITIAL='-----'	00018500
*			00018600
SRCH	DFHMDF	SIZE=(24,80),LINE=1,COLUMN=1	00018700
	DFHMDF	POS=(4,10),LENGTH=78,ATTRB=(BRT,PROT),PS=X'F8',	*00018800
*HG/23		INITIAL='RK RN GHKR DLS VUS'	00018900

	DFHMDF POS=(20,10),LENGTH=4,ATTRB=(PROT,BRT),INITIAL='---->'	00019000
SEGU	DFHMDF POS=(20,16),LENGTH=12,ATTRB=(IC,FSET), INITIAL='-----'	*00019100 00019200
*		00019300
UPDA	DFHMDF POS=(4,10),LENGTH=78,ATTRB=(BRT,PROT),PS=X'F8', INITIAL='RK RN TN WJD VUS'	00019400 *00019500
*HG/23	DFHMDF POS=(20,10),LENGTH=4,ATTRB=(PROT,BRT),INITIAL='---->'	00019600
UPDT	DFHMDF POS=(20,16),LENGTH=12,ATTRB=(IC,FSET), INITIAL='-----'	*00019700 *00019800 00019900
*		00020000
DELE	DFHMDF POS=(4,10),LENGTH=78,ATTRB=(BRT,PROT),PS=X'F8', INITIAL='RK RN TKR WP VUS'	00020100 *00020200
*HG/26	DFHMDF POS=(11,14),LENGTH=54,ATTRB=PROT,PS=X'F8', INITIAL='RKRNTKRWPTLDPS RKRNDNJSQJSGH 00 "KEY IN"'	00020300 *00020400
*HG/40	DFHMDF POS=(20,10),LENGTH=4,ATTRB=(PROT,BRT),INITIAL='---->'	00020500
DELT	DFHMDF POS=(20,16),LENGTH=12,ATTRB=(IC,FSET), INITIAL='-----'	*00020600 *00020700 00020800
*		00020900
MOVE	DFHMDF POS=(6,20),LENGTH=78,ATTRB=(BRT,PROT),PS=X'F8', INITIAL='RK RN DL EHD VUS'	00021000 *00021100
*HG/26	DFHMDF POS=(13,10),LENGTH=4,ATTRB=(PROT,BRT),INITIAL='---->'	00021200
MVE1	DFHMDF POS=(13,16),LENGTH=12,ATTRB=(IC,FSET), INITIAL='-----'	*00021300 *00021400 00021500
	DFHMDF POS=(15,10),LENGTH=4,ATTRB=(PROT,BRT),INITIAL='---->'	00021600
MVE2	DFHMDF POS=(15,16),LENGTH=12,ATTRB=(FSET), INITIAL='-----'	*00021700 00021800 00021900
*		00022000
PASS	DFHMDF POS=(9,10),LENGTH=40,ATTRB=(BRT,PROT), INITIAL='*****'	*00022100 00022200
	DFHMDF POS=(10,20),LENGTH=40,ATTRB=(BRT,PROT),PS=X'F8', INITIAL='QLALFQJSGH "KEY IN"'	*00022300 00022400
*HG/20	DFHMDF POS=(11,10),LENGTH=40,ATTRB=(BRT,PROT), INITIAL='*****'	*00022500 00022600
	DFHMDF POS=(13,10),LENGTH=6,ATTRB=(PROT,BRT), INITIAL='----->'	*00022700 00022800
PASKEY	DFHMDF POS=(13,18),LENGTH=4,ATTRB=(UNPROT,DRK,IC)	00022900
*		00023000
	DFHMDF TYPE=FINAL	00023100
	END	00023200
/*		00023300
//		00023400

A33,86.055,17:31:19,A33.PROGRAM.LIBRARY(CICSPLI1)

```
//A33VHFLI JOB CLASS=A,MSGCLASS=X,NOTIFY=A33
// EXEC DFHEITPL
//TRN.SYSIN DD *
* PROCESS S,GS,FLOW(100,10),ASSEMBLER ;
KANGPL1 : PROC OPTIONS(MAIN,REENTRANT) ;
%INCLUDE KKHMAP;
DCL 1 SEEN BASED(ADDR(SCREI)) UNALIGNED,
2 FILL CHAR(118),
2 REUSE(7),
3 NUML FIXED BINARY(15,0),
3 F1 CHAR(5),
3 NUMI CHAR(2),
3 NAMEL FIXED BINARY(15,0),
3 F2 CHAR(5),
3 NAMEI CHAR(6),
3 RELL FIXED BINARY(15,0),
3 F3 CHAR(5),
3 RELI CHAR(4),
3 SEXL FIXED BINARY(15,0),
3 F4 CHAR(5),
3 SEXI CHAR(1),
3 BIRL FIXED BINARY(15,0),
3 F5 CHAR(5),
3 BIRI CHAR(7),
3 HAKL FIXED BINARY(15,0),
3 F6 CHAR(5),
3 HAKI CHAR(3),
3 BAEL FIXED BINARY(15,0),
3 F7 CHAR(5),
3 BAEI CHAR(1),
3 CHOL FIXED BINARY(15,0),
3 F8 CHAR(5),
3 CHOI CHAR(4),
3 CHLL FIXED BINARY(15,0),
3 F9 CHAR(5),
3 CHLI CHAR(2),
3 IMSNL FIXED BINARY(15,0),
3 F10 CHAR(5),
3 IMSNI CHAR(5),
3 JUNL FIXED BINARY(15,0),
3 F11 CHAR(5),
3 JUNI CHAR(6),
3 BIGOL FIXED BINARY(15,0),
3 F12 CHAR(5),
3 BIGOI CHAR(1),
3 CHGAL FIXED BINARY(15,0),
3 F13 CHAR(5),
3 CHGAI CHAR(5);
DCL 1 IN_RECORD,
2 WCARD CHAR(1),
2 WILNO CHAR(5),
2 WKEY CHAR(12),
2 WNUM CHAR(2),
2 GITA CHAR(80);
DCL 1 TAIL1 DEF GITA,
2 WGEO CHAR(4),
2 WBUJUK CHAR(2),
2 WHANG CHAR(6),
2 WJUSD CHAR(40),
2 WF21 CHAR(1),
2 WGBGO CHAR(1),
2 WF22 CHAR(26);
DCL 1 TAIL2 DEF GITA,
```

```
00000100
00000200
00000300
00000400
00000500
00000600
00000700
00000800
00000900
00001000
00001100
00001200
00001300
00001400
00001500
00001600
00001700
00001800
00001900
00002000
00002100
00002200
00002300
00002400
00002500
00002600
00002700
00002800
00002900
00003000
00003100
00003200
00003300
00003400
00003500
00003600
00003700
00003800
00003900
00004000
00004100
00004200
00004300
00004400
00004500
00004600
00004700
00004800
00004900
00005000
00005100
00005200
00005300
00005400
00005500
00005600
00005700
00005800
00005900
00006000
00006100
00006200
00006300
```

2	WNAME	CHAR(20),	00006400
2	WREL	CHAR(4),	00006500
2	WSEX	CHAR(1),	00006600
2	WBIR	CHAR(7),	00006700
2	WHAK	CHAR(3),	00006800
2	WBAE	CHAR(1),	00006900
2	WCHO	CHAR(4),	00007000
2	WCHL	CHAR(2),	00007100
2	WIMSN	CHAR(5),	00007200
2	WJUN	CHAR(6),	00007300
2	WRIGO	CHAR(1),	00007400
2	WF4	CHAR(4),	00007500
2	WOFF	CHAR(2),	00007600
2	WCHGA	CHAR(5),	00007700
2	WCHGB	CHAR(5),	00007800
2	WCHGC	CHAR(5),	00007900
2	WCHGD	CHAR(5),	00008000
DCL	SILNO	CHAR(5),	00008100
	SVKEY	CHAR(12),	00008200
	SGEO	CHAR(4),	00008300
	SBUJUK	CHAR(2),	00008400
	SHANG	CHAR(6),	00008500
	SJUSO	CHAR(40) ;	00008600
DCL	SKEY	CHAR(12) INIT(' ');	00008700
DCL	TKEY	CHAR(14) INIT(' ');	00008800
DCL	MKEY	CHAR(14) INIT(' ');	00008900
DCL	MSG	CHAR(50) INIT(' ');	00009000
DCL	LEN	PIC '99' ;	00009100
DCL	FLEN	PIC '999' ;	00009200
DCL	(I,J)	PIC '999' INIT(0);	00009300
	LEN	= 50 ;	00009400
	FLEN	= 100 ;	00009500
		/* START */	00009600
EXEC	CICS HANDLE CONDITION	NOTFND(NOTFOUND)	00009700
		MAPFAIL(MAPFAIL)	00009800
		DUPREC(DUPREC)	00009900
		ENDFILE(ENDFILE) ;	00010000
BEGIN :			00010100
	INFMO = ' ' ;		00010200
EXEC	CICS SEND MAP('MENU') MAPSET('KKHMAP') ERASE ;		00010300
EXEC	CICS RECEIVE MAP('MENU') MAPSET('KKHMAP') ;		00010400
	IF INFMI = '1' THEN GOTO UPDATE_RTN ;		00010500
	IF INFMI = '2' THEN GOTO UPDATE_RTN ;		00010600
	IF INFMI = '3' THEN GOTO INSERT_RTN ;		00010700
	IF INFMI = '4' THEN GOTO DEL_RTN ;		00010800
	IF INFMI = '5' THEN GOTO MOVE_RTN ;		00010900
	IF INFMI = '6' THEN GOTO LAST_RTN ;		00011000
	GO TO BEGIN ;		00011100
UPDATE_RTN :			00011200
	IF INFMI='1' THEN		00011300
DO,	SEGUO = '?.....?';		00011400
	EXEC CICS SEND MAP('SRCH') MAPSET('KKHMAP') ERASE ;		00011500
	EXEC CICS RECEIVE MAP('SRCH') MAPSET('KKHMAP') ;		00011600
	TKEY=SEGUI ;		00011700
END ;			00011800
	IF INFMI='2' THEN		00011900
DO,	UPDTO = '?.....?';		00012000
	EXEC CICS SEND MAP('UPDA') MAPSET('KKHMAP') ERASE ;		00012100
	EXEC CICS RECEIVE MAP('UPDA') MAPSET('KKHMAP') ;		00012200
	TKEY=UPDTI ;		00012300
END ;			00012400
STAR	EXEC CICS STARTBR DATASET('FILEK') RIDFLD(TKEY) GTERQ ;		00012500
	EXEC CICS READNEXT DATASET('FILEK') INTO(IN_RECORD) RIDFLD(TKEY);		00012600



SKEY=TKEY ;	00012700
ILNOO=' ' ; JONOO=' ' ; GANOO=' ' ; GEOO=' ' ; BUJUKO=' ' ;	00012800
HANGO=' ' ; JUSOO=' ' ;	00012900
LOOP:CALL CLEAR_RTN ;	00013000
J=0 ;	00013100
DO WHILE(WKEY=SKEY & J < 7) ;	00013200
IF WNUM='00' THEN CALL SAVE_RTN ;	00013300
ELSE DO ; J=J+1 ;	00013400
CALL GAGUWON_RTN ;	00013500
END ;	00013600
EXEC CICS READNEXT DATASET('FILEK') INTO(IN_RECORD)	00013700
RIDFLD(TKEY) ;	00013800
END ;	00013900
CALL GAGU_RTN ;	00014000
SDMAP:INCRO=' ' ;	00014100
EXEC CICS SEND MAP('SCRE') MAPSET('KKHMAP') ERASE ;	00014200
EXEC CICS RECEIVE MAP('SCRE') MAPSET('KKHMAP') ;	00014300
EXEC CICS ENDBR DATASET('FILEK') ;	00014400
SELECT(INCRI) ;	00014500
WHEN('F')	00014600
DO ;	00014700
IF INFMI = '2' THEN CALL MODIFY_RTN ;	00014800
IF I > 7 THEN	00014900
DO ; TKEY = SKEY  NUMI(7) ;	00015000
EXEC CICS STARTBR DATASET('FILEK') RIDFLD(TKEY) GTEQ ;	00015100
EXEC CICS READNEXT DATASET('FILEK') INTO(IN_RECORD)	00015200
RIDFLD(TKEY) ;	00015300
GO TO LOOP ;	00015400
END ;	00015500
ELSE	00015600
DO ;	00015700
IF INFMI = '1' THEN DO ;	00015800
TKEY = WKEY  WNUM ;	00015900
GOTO STAR ;	00016000
END ;	00016100
ELSE GOTO UPDATE_RTN ;	00016200
END ;	00016300
END ;	00016400
WHEN('Q') GOTO UPDATE_RTN ;	00016500
WHEN('B') GOTO BACK_RTN ;	00016600
OTHERWISE GOTO UPDATE_RTN ;	00016700
END ;	00016800
MSG = ' UPDATE COMPLETED ..... PRESS ENTER TO CONTINUE' ;	00016900
EXEC CICS SEND FROM(MSG) LENGTH(50) ERASE ;	00017000
EXEC CICS RECEIVE INTO(MSG) LENGTH(LEN) ;	00017100
GO TO BEGIN ;	00017200
BACK_RTN :	00017300
IF NUMI(1) = '01' THEN	00017400
DO ;	00017500
IF INFMI = '1' THEN	00017600
DO ;	00017700
TKEY=SKEY  '00' ;	00017800
EXEC CICS STARTBR DATASET('FILEK') RIDFLD(TKEY) GTEQ ;	00017900
EXEC CICS READPREV DATASET('FILEK') RIDFLD(TKEY)	00018000
INTO(IN_RECORD) ;	00018100
EXEC CICS READPREV DATASET('FILEK') RIDFLD(TKEY)	00018200
INTO(IN_RECORD) ;	00018300
EXEC CICS ENDBR DATASET('FILEK') ;	00018400
TKEY=WKEY  '00' ;	00018500
GOTO STAR ;	00018600
END ;	00018700
ELSE GOTO UPDATE_RTN ;	00018800
END ;	00018900

ELSE DO ;	00019000
TKEY = SKEY  NUMI(1) ;	00019100
EXEC CICS STARTBR DATASET('FILEK') RIDFLD(TKEY) GTEQ ;	00019200
EXEC CICS READPREV DATASET('FILEK') RIDFLD(TKEY)	00019300
INTO(IN_RECORD) ;	00019400
DO J = 7 TO 1 BY -1 ;	00019500
EXEC CICS READPREV DATASET('FILEK') RIDFLD(TKEY)	00019600
INTO(IN_RECORD) ;	00019700
CALL GAGUWON_RTN ;	00019800
END ;	00019900
GOTO SDMAP ;	00020000
END ;	00020100
GO TO BEGIN ;	00020200
MODIFY_RTN : PROC ;	00020300
TKEY = SKEY  '00' ;	00020400
EXEC CICS READ UPDATE DATASET('FILEK')	00020500
INTO(IN_RECORD) RIDFLD(TKEY) ;	00020600
CALL CGAGU_RTN ;	00020700
EXEC CICS REWRITE FROM(IN_RECORD) DATASET('FILEK') ;	00020800
DO I = 1 TO J ;	00020900
TKEY = SKEY  NUMI(I) ;	00021000
EXEC CICS READ UPDATE DATASET('FILEK') INTO(IN_RECORD)	00021100
RIDFLD(TKEY) ;	00021200
CALL CGAGUWON_RTN ;	00021300
EXEC CICS REWRITE FROM(IN_RECORD) DATASET('FILEK') ;	00021400
END ;	00021500
DO I = J+1 TO 7 WHILE(NUMI(I) ^= ' ') ;	00021600
IN_RECORD = ' ' ;	00021700
IF NUML(I) > 0 THEN	00021800
DO; TKEY = SKEY  NUMI(I) ;	00021900
WCARD = '1' ;	00022000
WILNO = ILNOI ;	00022100
WKEY = SKEY ;	00022200
WNUM = NUMI(I) ;	00022300
CALL CGAGUWON_RTN ;	00022400
EXEC CICS WRITE FROM(IN_RECORD) DATASET('FILEK')	00022500
RIDFLD(TKEY) ;	00022600
END ;	00022700
END ;	00022800
END MODIFY_RTN ;	00022900
DEL_RTN :	00023000
DELTO = '?.....?';	00023100
EXEC CICS SEND MAP('DELE') MAPSET('KKHMAP') ERASE ;	00023200
EXEC CICS RECEIVE MAP('DELE') MAPSET('KKHMAP') ;	00023300
IF SUBSTR(DELTO,13,2) = '00' THEN	00023400
DO; EXEC CICS DELETE DATASET('FILEK') RIDFLD(DELTI)	00023500
GENERIC KEYLENGTH(12) ;	00023600
END; ELSE	00023700
DO ;	00023800
EXEC CICS READ DATASET('FILEK') INTO(IN_RECORD)	00023900
RIDFLD(DELTI) UPDATE ;	00024000
EXEC CICS DELETE DATASET('FILEK') ;	00024100
END ;	00024200
MSG = 'DELETE COMPLETED ..... ENTER TO CONTINUE ' ;	00024300
EXEC CICS SEND FROM(MSG) LENGTH(50) ERASE ;	00024400
EXEC CICS RECEIVE INTO(MSG) LENGTH(LEN) ;	00024500
GO TO BEGIN ;	00024600
INSERT_RTN :	00024700
ILNOO=' ' ; JONOO=' ' ; GANOO=' ' ; GEOO=' ' ; BUJUKO=' ' ;	00024800
HANGO=' ' ; JUSOO=' ' ;	00024900
CALL CLEAR_RTN ;	00025000
ADDO = '?.....?';	00025100
	00025200

```

EXEC CICS SEND MAP('INST') MAPSET('KKHMAP') ERASE ; 00025300
EXEC CICS RECEIVE MAP('INST') MAPSET('KKHMAP') ; 00025400
INCR1=' ' ; 00025500
EXEC CICS SEND MAP('SCRE') MAPSET('KKHMAP') ERASE ; 00025600
EXEC CICS RECEIVE MAP('SCRE') MAPSET('KKHMAP') ; 00025700
W CARD = '1' ; 00025800
WKEY = ADDI ; 00025900
WNUM = '00' ; 00026000
TKEY = ADDI||'00' ; 00026100
CALL CGAGU_RTN ; 00026200
EXEC CICS WRITE FROM(IN_RECORD) DATASET('FILEK') RIDFLD(TKEY) ; 00026300
DO I = 1 TO 7 WHILE( NAMEI(I) = ' ' ) ; 00026400
WILNO = ILNOI ; 00026500
W CARD = '1' ; 00026600
WKEY = ADDI ; 00026700
WNUM = NUMI(I) ; 00026800
TKEY = ADDI||NUMI(I) ; 00026900
CALL CGAGUWON_RTN ; 00027000
EXEC CICS WRITE FROM(IN_RECORD) DATASET('FILEK') RIDFLD(TKEY) ; 00027100
END ; 00027200
DLOOP: 00027300
SELECT(INCR1) ; 00027400
WHEN('F') 00027500
DO ; 00027600
IF I > 7 THEN DO ; 00027700
EXEC CICS SEND MAP('SCRE') MAPSET('KKHMAP') ERASE ; 00027800
EXEC CICS RECEIVE MAP('SCRE') MAPSET('KKHMAP') ; 00027900
W CARD = '1' ; 00028000
WILNO = ILNOI ; 00028100
WKEY = ADDI ; 00028200
WNUM = NUMI(I) ; 00028300
TKEY = ADDI||NUMI(I) ; 00028400
CALL CGAGUWON_RTN ; 00028500
EXEC CICS WRITE FROM(IN_RECORD) DATASET('FILEK') 00028600
RIDFLD(TKEY) ; 00028700
GO TO DLOOP ; 00028800
END ; 00028900
END ; 00029000
OTHERWISE ; 00029100
END ; 00029200
MSG = ' INSERT COMPLETED ..... PRESS ENTER TO CONTINUE ' ; 00029300
EXEC CICS SEND FROM(MSG) LENGTH(50) ERASE ; 00029400
EXEC CICS RECEIVE INTO(MSG) LENGTH(LEN) ; 00029500
GO TO BEGIN ; 00029600
MOVE_RTN : 00029700
MVE10 = '?.....?'; 00029800
MVE20 = '?.....?'; 00029900
EXEC CICS SEND MAP('MOVE') MAPSET('KKHMAP') ERASE ; 00030000
EXEC CICS RECEIVE MAP('MOVE') MAPSET('KKHMAP') ; 00030100
TKEY=MVE1I ; 00030200
MSTAR:EXEC CICS STARTBR DATASET('FILEK') RIDFLD(TKEY) GTER ; 00030300
EXEC CICS READNEXT DATASET('FILEK') INTO(IN_RECORD) 00030400
RIDFLD(TKEY) 00030500
SKEY=MVE1I ; 00030600
ILNOO=' ' ; JONOO=' ' ; GANOO=' ' ; GEOO=' ' ; BUJUKO=' ' ; 00030700
HANGO=' ' ; JUSOO=' ' ; 00030800
MLOOP:CALL CLEAR_RTN ; 00030900
J = 0 ; 00031000
DO WHILE(WKEY=SKEY & J < 7) ; 00031100
IF WNUM='00' THEN CALL SAVE_RTN ; 00031200
ELSE DO ; J=J+1 ; 00031300
CALL GAGUWON_RTN ; 00031400
END ; 00031500

```

EXEC CICS READNEXT DATASET('FILEK') INTO(IN_RECORD)	00031600
RIDFLD(TKEY) ;	00031700
END;	00031800
CALL GAGU_RTN;	00031900
MVMAP:INCRO=' ' ;	00032000
EXEC CICS SEND MAP('SCRE') MAPSET('KKHMAP') ERASE;	00032100
EXEC CICS RECEIVE MAP('SCRE') MAPSET('KKHMAP') ;	00032200
EXEC CICS ENDBR DATASET('FILEK') ;	00032300
TKEY = SKEY   '00' ;	00032400
MKEY = MVE2I  '00' ;	00032500
EXEC CICS READ UPDATE DATASET('FILEK')	00032600
INTO(IN_RECORD) RIDFLD(TKEY) ;	00032700
CALL CGAGU_RTN ;	00032800
IF WBG0 = ' ' THEN WBG0 = '8' ;	00032900
EXEC CICS REWRITE FROM(IN_RECORD) DATASET('FILEK') ;	00033000
WKEY = MVE2I ;	00033100
WNUM = '00' ;	00033200
WBG0 = '3' ;	00033300
EXEC CICS WRITE FROM(IN_RECORD) DATASET('FILEK') RIDFLD(MKEY);	00033400
DO I = 1 TO 7 WHILE(NAMEI(I) ^= ' ') ;	00033500
TKEY = SKEY   NUMI(I) ;	00033600
MKEY = MVE2I  NUMI(I) ;	00033700
EXEC CICS READ UPDATE DATASET('FILEK') INTO(IN_RECORD)	00033800
RIDFLD(TKEY) ;	00033900
IF NAMEI(I) > 0 THEN WNAME = NAMEI(I);	00034000
IF RELL(I) > 0 THEN WREL = RELI(I);	00034100
IF SEXL(I) > 0 THEN WSEX = SEXI(I);	00034200
IF BIRL(I) > 0 THEN WBIR = BIRI(I);	00034300
IF HAKL(I) > 0 THEN WHAK = HAKI(I);	00034400
IF BAEL(I) > 0 THEN WBAE = BAEI(I);	00034500
IF CHOL(I) > 0 THEN WCHO = CHOI(I);	00034600
IF CHLL(I) > 0 THEN WCHL = CHLI(I);	00034700
IF IMSNL(I) > 0 THEN WIMSN = IMSNI(I);	00034800
IF JUNL(I) > 0 THEN WJUN = JUNI(I);	00034900
IF CHGAL(I) > 0 THEN WCHGD = CHGAI(I);	00035000
IF WBIGO = ' ' THEN WBIGO = '8';	00035100
EXEC CICS REWRITE FROM(IN_RECORD) DATASET('FILEK') ;	00035200
WBIGO = '3' ;	00035300
WKEY = MVE2I ;	00035400
WNUM = NUMI(I) ;	00035500
EXEC CICS WRITE FROM(IN_RECORD) DATASET('FILEK') RIDFLD(MKEY);	00035600
END ;	00035700
SELECT(INCRI) ;	00035800
WHEN('F')	00035900
DO;	00036000
IF I > 7 THEN	00036100
DO; TKEY = SKEY  NUMI(7) ;	00036200
EXEC CICS STARTBR DATASET('FILEK') RIDFLD(TKEY) GTER ;	00036300
EXEC CICS READNEXT DATASET('FILEK') INTO(IN_RECORD)	00036400
RIDFLD(TKEY) ;	00036500
GO TO MLOOP ;	00036600
END ;	00036700
ELSE GOTO MOVE_RTN ;	00036800
END ;	00036900
WHEN('Q') GOTO MOVE_RTN ;	00037000
WHEN('B') GOTO MBACK_RTN ;	00037100
OTHERWISE GOTO MOVE_RTN ;	00037200
END ;	00037300
MSG = ' MOVE COMPLETED ..... PRESS ENTER TO CONTINUE' ;	00037400
EXEC CICS SEND FROM(MSG) LENGTH(50) ERASE ;	00037500
EXEC CICS RECEIVE INTO(MSG) LENGTH(LEN) ;	00037600
GO TO BEGIN ;	00037700
MBACK_RTN :	00037800

```

IF NUMI(1) = '01' THEN                                00037900
DO ;                                                    00038000
  IF INFMI = '1' THEN                                  00038100
  DO ;                                                  00038200
    TKEY=SKEY||'00' ;                                  00038300
    EXEC CICS STARTBR DATASET('FILEK') RIDFLD(TKEY) GTER ; 00038400
    EXEC CICS READPREV DATASET('FILEK') RIDFLD(TKEY)
      INTO(IN_RECORD) ;                                00038500
    EXEC CICS READPREV DATASET('FILEK') RIDFLD(TKEY)
      INTO(IN_RECORD) ;                                00038700
    EXEC CICS ENDBR DATASET('FILEK') ;                 00038800
    TKEY=WKEY||'00' ;                                  00038900
    GOTO MSTAR ;                                       00039000
  END ;                                                00039100
  ELSE GOTO MOVE_RTN ;                                  00039200
END ;                                                  00039300
ELSE DO ;                                             00039400
  TKEY = SKEY||NUMI(1) ;                               00039500
  EXEC CICS STARTBR DATASET('FILEK') RIDFLD(TKEY) GTER ; 00039600
  EXEC CICS READPREV DATASET('FILEK') RIDFLD(TKEY)
    INTO(IN_RECORD) ;                                00039700
  DO J = 7 TO 1 BY -1 ;                                00039800
    EXEC CICS READPREV DATASET('FILEK') RIDFLD(TKEY)
      INTO(IN_RECORD) ;                                00039900
    CALL GAGUWON_RTN ;                                 00040000
  END ;                                                00040100
  GOTO MVMAP ;                                         00040200
END ;                                                  00040300
GO TO BEGIN ;                                         00040400
CLEAR_RTN : PROC ;                                    00040500
  NUMI = ' ' ; NAMEI = ' ' ; RELI = ' ' ;              00040600
  SEXI = ' ' ; HAKI = ' ' ; BIRI = ' ' ; BAEI = ' ' ; CHOI = ' ' ; 00040700
  CHLI = ' ' ; IMSNI = ' ' ; JUNI = ' ' ; BIGOI = ' ' ; 00040800
  CHGAI = ' ' ;                                        00040900
END CLEAR_RTN ;                                       00041000
GAGU_RTN : PROC ;                                     00041100
  ILNOO = SILNO ;                                     00041200
  JONOO = SUBSTR(SVKEY,1,4) ;                          00041300
  GANOO = SUBSTR(SVKEY,5,8) ;                          00041400
  GEOO = SGEO ;                                       00041500
  BUJUKO = SBUJUK ;                                    00041600
  HANGO = SHANG ;                                      00041700
  JUSOO = SJUSO ;                                     00041800
END GAGU_RTN ;                                        00041900
SAVE_RTN : PROC ;                                    00042000
  SILNO = WILNO ;                                     00042100
  SUBSTR(SVKEY,1,4) = SUBSTR(WKEY,1,4) ;               00042200
  SUBSTR(SVKEY,5,8) = SUBSTR(WKEY,5,8) ;               00042300
  SGEO = WGEO ;                                       00042400
  SBUJUK = WBUJUK ;                                    00042500
  SHANG = WHANG ;                                     00042600
  SJUSO = WJUSO ;                                     00042700
END SAVE_RTN ;                                       00042800
GAGUWON_RTN : PROC ;                                  00042900
  NUMI(J) = WNUM ;                                    00043000
  NAMEI(J) = WNAME ;                                  00043100
  RELI(J) = WREL ;                                    00043200
  SEXI(J) = WSEX ;                                    00043300
  BIRI(J) = WBIR ;                                    00043400
  HAKI(J) = WHAK ;                                    00043500
  BAEI(J) = WBAE ;                                    00043600
  CHOI(J) = WCHO ;                                    00043700
  CHLI(J) = WCHL ;                                    00043800

```

```

        IMSNI(J)= WIMSN ;
        JUNI(J) = WJUN ;
        BIGOI(J) = WBIGO ;
        CHGAI(J) = WCHGD ;
END GAGUWON_RTN ;
CGAGU_RTN : PROC ;
    IF ILNOL > 0 THEN WILNO=ILNOI ;
    IF GEOL > 0 THEN WGEO = GEOI ;
    IF BUJUKL > 0 THEN WBUJUK = BUJUKI ;
    IF HANGL > 0 THEN WHANG = HANGI ;
    IF JUSOL > 0 THEN WJUSO = JUSOI ;
END CGAGU_RTN ;
CGAGUWON_RTN : PROC ;
    IF NAMEI(I) > 0 THEN WNAME = NAMEI(I) ;
    IF RELI(I) > 0 THEN WREL = RELI(I) ;
    IF SEXI(I) > 0 THEN WSEX = SEXI(I) ;
    IF BIRI(I) > 0 THEN WBIR = BIRI(I) ;
    IF HAKI(I) > 0 THEN WHAK = HAKI(I) ;
    IF BAEI(I) > 0 THEN WBAE = BAEI(I) ;
    IF CHOI(I) > 0 THEN WCHO = CHOI(I) ;
    IF CHLI(I) > 0 THEN WCHL = CHLI(I) ;
    IF IMSNI(I) > 0 THEN WIMSN = IMSNI(I) ;
    IF JUNI(I) > 0 THEN WJUN = JUNI(I) ;
    IF BIGOI(I) > 0 THEN WBIGO = BIGOI(I) ;
    IF CHGAI(I) > 0 THEN WCHGD = CHGAI(I) ;
END CGAGUWON_RTN ;
MAPFAIL :
    MSG = ' MAP NOT FOUND ..... ENTER TO CONTINUE ' ;
    EXEC CICS SEND FROM(MSG) LENGTH(50) ERASE ;
    EXEC CICS RECEIVE INTO(MSG) LENGTH(LEN) ;
    GO TO BEGIN ;
NOTFOUND :
    MSG = ' DATA NOT FOUND ..... ENTER TO CONTINUE ' ;
    EXEC CICS SEND FROM(MSG) LENGTH(50) ERASE ;
    EXEC CICS RECEIVE INTO(MSG) LENGTH(LEN) ;
    GO TO BEGIN ;
ENDFILE :
    MSG = ' FILE ENDED ..... ENTER TO CONTINUE ' ;
    EXEC CICS SEND FROM(MSG) LENGTH(50) ERASE ;
    EXEC CICS RECEIVE INTO(MSG) LENGTH(LEN) ;
    TKEY=WKEY||WNUM ;
    EXEC CICS RESETBR DATASET('FILEK') RIDFLD(TKEY) ;
    EXEC CICS ENDBR DATASET('FILEK') ;
    GO TO BEGIN ;
DUPREC :
    MSG = ' RECORD DUPLICATED .. ENTER TO CONTINUE ' ;
    EXEC CICS SEND FROM(MSG) LENGTH(50) ERASE ;
    EXEC CICS RECEIVE INTO(MSG) LENGTH(LEN) ;
    GO TO BEGIN ;
LAST_RTN :
    MSG='CICS ENDED ..... CSSF LOGOFF KEYIN!!!' ;
    EXEC CICS SEND FROM(MSG) LENGTH(50) ERASE ;
    EXEC CICS RETURN ;
END KANGPL1 ;
/*
//PLI.SYSLIB DD DSN=CICSDPD.MAPLIB,DISP=SHR
// DD DSN=CICS.PL1LIB,DISP=SHR
//LKED.SYSLMOD DD DSN=CICSDPD.LOADLIB,DISP=SHR
//LKED.SYSIN DD *
        NAME KANGPL1(R)
/*
//

```

```

00044200
00044300
00044400
00044500
00044600
00044700
00044800
00044900
00045000
00045100
00045200
00045300
00045400
00045500
00045600
00045700
00045800
00045900
00046000
00046100
00046200
00046300
00046400
00046500
00046600
00046700
00046800
00046900
00047000
00047100
00047200
00047300
00047400
00047500
00047600
00047700
00047800
00047900
00048000
00048100
00048200
00048300
00048400
00048500
00048600
00048700
00048800
00048900
00049000
00049100
00049200
00049300
00049400
00049500
00049600
00049700
00049800
00049900
00050000
00050100
00050200
00050300

```

A33,86.147,11:56:02,A33.PROGRAM.LIBRARY(BACKUPT)

```
//A33VHWVT JOB CLASS=A,MSGCLASS=X,NOTIFY=A33
//JOB CAT DD DSN=CATALOG.VBOS005,DISP=SHR
//STEP1 EXEC PLIFCLG
//PLI.SYSPRINT DD SYSOUT=*
//PLI.SYSIN DD *
* PROCESS GS,A,XREF,OPT(2);
  RIVER: PROC OPTIONS(MAIN);
    DCL DISKIN FILE RECORD INPUT ENV(VSAM);
    DCL TAPEOUT FILE RECORD OUTPUT;
    DCL 1 TT CHAR(100);
      DCL 1 TA DEF TT,
        2 CARD CHAR(6),
        2 JOSA PIC '(4)9',
        2 KEYA CHAR(10),
        2 GU CHAR(6),
        2 ADDR CHAR(40),
        2 FILA CHAR(34);
      DCL 1 TB DEF TT,
        2 CARD1 CHAR(6),
        2 JOSA1 PIC '(4)9',
        2 KEYA1 CHAR(10),
        2 NAME CHAR(20),
        2 NUM CHAR(33),
        2 BLK CHAR(27);
    DCL TPO FILE RECORD ENV(VSAM);
      DCL 1 TTT CHAR(100);
        DCL 1 TAA DEF TTT,
          2 CARD CHAR(6),
          2 JOSA PIC '(4)9',
          2 KEYA CHAR(10),
          2 GU CHAR(6),
          2 HJ PIC '(6)9',
          2 OADDR CHAR(40),
          2 FIL1 CHAR(28);
        DCL 1 TBB DEF TTT,
          2 CARD1 CHAR(6),
          2 JOSA1 PIC '(4)9',
          2 KEYA1 CHAR(10),
          2 ONAME CHAR(20),
          2 NUM CHAR(33),
          2 BLK CHAR(27);
    DCL HGBATCH ENTRY OPTIONS(ASSEMBLER);
    DCL LA FIXED BIN(15) INIT(40);
    DCL LB FIXED BIN(15) INIT(40);
    DCL LC FIXED BIN(15) INIT(16);
    DCL LD FIXED BIN(15) INIT(10);
    DCL FUNC CHAR(2) INIT('IF');
    DCL ABC CHAR(20) INIT('');
    DCL DDD CHAR(40) INIT('');
    DCL RECODE CHAR(2) INIT(' ');
    DCL (OCNT,ICNT,ECNT) PIC '(6)9' INIT(0);
    DCL DJOSA(570) PIC '(4)9' INIT
(0001,0002,0003,0005,0006,0007,0009,0010,1011,2011,0013,0014,
0015,0017,0018,0019,0021,0022,0023,0025,0026,0027,0029,0030,
0031,0033,0034,0035,0037,0038,0039,0041,0042,0043,0045,0046,
0047,0049,0050,0051,0053,0054,0055,0057,0058,0059,0061,0062,
0063,0065,
0066,0067,0069,0070,0071,0073,0074,0075,0077,0078,
0079,0081,0082,0083,0085,0086,0087,0089,0090,0091,
0093,0094,0095,0097,0098,0099,0101,0102,0103,0105,
0106,0107,0109,0110,0111,0113,0114,0115,0117,0118,
0119,0121,0122,0123,0125,0126,0127,0129,0130,0131,
0133,0134,0135,0137,0139,0141,0142,0143,0145,0147,
```

```

0149,0150,0151,0153,0154,0155,0157,0158,0159,0161,
0162,1163,2163,0165,0166,0167,0169,0170,0171,0173,
0174,0175,0177,0178,0179,0181,0182,0183,0185,0186,
0187,0189,0190,0191,0193,0194,0195,0197,0198,0199,
0201,0202,0203,0205,0206,0209,0210,0211,0213,0214,
0215,0217,0218,0219,0221,0222,0223,0225,0226,0227,
0229,0230,0250,0251,0253,0254,0255,0257,0258,0259,
0261,0262,0263,0265,0266,0267,0269,0270,0271,0273,
0274,0275,0277,0278,0279,0281,0282,0283,0285,0286,
0287,0289,0290,0291,0293,0294,0295,0297,0298,0299,
0301, 1302,2302,3302,
0303,0305,0306,0307,0309,0310,0311,0313,
0314,0315,0317,0318,0319,0321,0322,0323,0325,0326,
0327,0329,0330,0331,0333,0334,0335,0337,0338,0351,0353,
0354,0355,0357,0358,0359,0361,0362,0363,0365,0366,
0367,0369,0370,0371,0373,0374,0375,0377,0378,0379,
0381,0382,0383,0385,0386,0387,0389,0390,0391,0393,
0394,0395,0397,0398,0399,0401,0403,0405,0406,0407,
0409,0410,0411,0413,0414,0415,0417,0418,0419,0421,
0422,0423,0425,0426,0427,0429,0430,0431,0433,0434,
0502,0503,0505,0506,0507,0509,0510,0511,0513,0514,
0515,0517,0518,0519,0521,0522,0523,0525,0526,0527,
0529,0530,0531,0533,0534,1535,2535,3535,4535,0537,
0538,1539,2539,0541,0542,0543,0545,0546,0547,0549,0550,
0551,0553,0554,0555,0557,0558,0559,0561,0562,0563,
0565,0566,0567,0571,0572,0573,1574,2574,0575,0576,0577,
0578,0579,0580,0581,0582,1583,2583,3583,0584,0589,
0590,0591,0592,0593,0594,0595,0603,0605,0606,
1607,2607,3607,
0609,0610,0611,0613,0614,0618,0619,0621,0622,0623,0625,
0626,0627,0629,0630,0631,0633,0634,0635,0671,0672,
0673,0674,0675,0676,0680,0681,0682,0683,0684,0685,
0686,0687,0688,0689,0690,0691,0692,0701,0702,0703,
0705,0706,0707,0709,0710,1711,2711,3711,0713,0714,
0715,0717,0718,0719,0725,0726,0727,0729,0730,0731,
0733,0734,0735,0737,0738,0739,0741,0742,0743,0745,
0746,0747,0749,0750,0751,0753,0754,0755,0771,0772,
0773,0774,0775,0776,0777,0778,0779,0783,0784,0785,
0786,0787,0788,0789,0790,0791,0792,0793,0794,0795,
0796,1803,2803,0805,0806,0807,0809,0810,0811,0813,
0814,0815,0817,0818,0819,0821,0822,0827,0829,0830,
0831,0833,0834,0835,0837,0838,0839,0841,0842,0843,
0845,0846,0847,0849,0850,0851,0853,0854,0855,0858,
0859,0861,0862,0863,0865,0866,0867,0869,0871,0872,
0873,0874,0875,0876,0877,0878,0879,0880,0881,0882,
0883,0884,0885,0886,0890,0891,0892,0893,0894,0895,
0896,0897,0898,0899,0981,0982,
0903,0905,0906,0907,0992,0993) ;
DCL HJCHK(570) PIC '(6)9' INIT(
111735,112634,112635,111542,112718,111544,111553,112640,112125,112125,00011300
111223,112611,112535,111639,112025,111417,111412,111513,112028,111631,00011400
111919,111513,111616,112027,111722,111531,112324,111816,112030,111822,00011500
112139,111512,112236,111630,111917,112319,112024,112524,112020,112529,00011600
111116,111716,111545,111222,111919,112331,111515,111728,111228,111627,00011700
112518,111534,111539,112020,112517,112122,111114,111616,111813,112031,00011800
111430,111548,111538,111421,111514,111723,112034,111128,112422,112237,00011900
111227,111439,111420,111552,111228,111728,112212,112526,111826,111629,00012000
112121,111711,112414,111811,112319,111550,111430,112028,111522,112323,00012100
111520,111416,111412,112517,112224,111550,112527,111930,111540,111926,00012200
112639,112411,112729,111630,111636,112022,111620,112613,112529,112514,00012300
111217,111927,111621,112119,111124,112531,112425,111325,111441,111627,00012400
111635,112621,112621,112526,111739,111613,111742,112011,111515,111226,00012500
112125,111746,111734,112114,112716,112630,112612,111415,111632,112723,00012600

```



```

112633,112314,111121,112715,112729,112615,112715,112617,112630,112630, 00012700
112629,112617,111631,112217,111435,112320,111222,111318,111518,112416, 00012800
112323,112229,111726,112318,111631,111219,111428,112235,111816,112331, 00012900
112314,112719,211227,211828,211228,212111,212111,211819,211424,212022, 00013000
211511,211425,211538,211221,211630,211523,211114,212013,211619,211526, 00013100
211615,212022,211615,211729,211626,211643,211225,211537,211328,211638, 00013200
211734,211320,211647,211623,212017,211617,211516,211411,211614,211823, 00013300
211617, 211654,211654,211654,211720,211619,211619, 00013400
211535,211629,211734,211827,211816,211726,211638,211712,211637,211922, 00013500
211223,212013,211533,211617,211820,211424,211415,211624,211526,211718, 00013600
211818,211716,211424,212018, 00013700
221628,221233,221222,221327,221224,221326,221334, 00013800
221322,221412,221215,221514,221211,221320,221217,221128,221523,221521, 00013900
221611,221214,221126,221321,221330,221523,221411,221328,221129,221217, 00014000
221215,221334,221316,221516,221417,221516,221325,221211,221337,221521, 00014100
221514,231219,231423,231436,231320,231415,231220,231116,231120,231214, 00014200
231318,231313,231314,231329,231413,231323,231417,231219,231212,231327, 00014300
231428,231318,231424,231419,231316,311712,311713,311114,311129,311121, 00014400
311115,311128,311238,311218,311224,311218,311213,311211, 00014500
311228,311220,311227,311320,311319,311313,311427,311413,311414, 00014600
311412,311427,311520,311520,311520,311520,311511,311518,311513,311513, 00014700
311615,311611,311611,311620,311411,311427, 00014800
321128,321117,321111,321219,321220, 00014900
321223,321416,321420,321525,321417,321321,321322,321312,321612,321122, 00015000
313411, 00015100
313912,313712,314511,314511,312215,314535,314434,314737,313537,314633, 00015200
314137,314339,311915,311915,311915,313631,324411,323713,323311,323732, 00015300
323133,323432,321622,331128,331116,331118, 331125,331125,331125, 00015400
331219,331213,331314, 00015500
331313,331121, 00015600
340115,340222,340112,340130,340126,340214,340117,340224,340232, 00015700
340212,340231, 00015800
340229,341121,341112,333411,333931,333831,333636,334034,333138,344112, 00015900
343211,343111,341311,343536,344331,343736,344540,344433,344242,344032, 00016000
343137,343432,351143,351137,351127,351120,351117,351123,351126,351418, 00016100
351226,351226,351226,351412,351511,351217,351415,351312,351216,360231, 00016200
360115,360211,360122,260317,360112,360332,360220,360213,360119,360121, 00016300
360314,360119, 360325,361113,361137,361132,361117, 00016400
361122,361234,361316,361225,361213,361112,353411,353112,354334,353633, 00016500
354234,354035,353134,353841,354136,364011,364811,363211,364211,363731, 00016600
363938,364934,364133,364531,363134,363536,364641,364432,364042,371121, 00016700
371121,371133,371114,371122,371111,371223,371228,371211,371319,371519, 00016800
371320,371314,371423,371414,371420,381111,381119,381117,381120,381135, 00016900
381134,381116,381516,381146,381128,381143,380311,380223,380213,380213, 00017000
380124,380217,380116,380215,381718,381820,381423,381331,381325,381323, 00017100
381327,381612,381618,381617,381423,381427,374912,371814,373411,375311, 00017200
373912,374332,374734,374432,373343,374536,373731,373440,374034,371322, 00017300
375131,374032,383412,383612,383735,384431,384834,383334,384737,384242, 00017400
384336,383836,383146,384537,391121,391125,391114,391117,393111,393131 00017500
) ; 00017600
DCL INREC CHAR(100); 00017700
DCL MM CHAR(100); 00017800
DCL (ICN,WCN) PIC'(6)9' INIT(0); 00017900
ON ENDFILE(DISKIN) GO TO CLOS; 00018000
ON ENDFILE(SYSIN) GO TO BEGIN; 00018100
OPEN FILE(DISKIN),FILE(TAPEOUT), 00018200
FILE(SYSPRINT) LINESIZE(132); 00018300
RD GET SKIP EDIT(MM) (A(100)); 00018400
SUBSTR(TT,1,100)=MM; 00018500
CALL COV_R; 00018600
WRITE FILE(TAPEOUT) FROM(TTT); 00018700
GO TO RD; 00018800
BEGIN: READ FILE(DISKIN) INTO(INREC); 00018900

```

```

ICN = ICN + 1 ;
IF SUBSTR(INREC,7,12) = '005806010103' THEN DO;
    SUBSTR(INREC,7,12) = '005806010102' ;
    CALL PRT_R ;
    END ;
ELSE IF SUBSTR(INREC,7,6) = '068201' &
    SUBSTR(INREC,74,1) < '5' THEN DO;
    SUBSTR(INREC,74,1) = 'D' ;
    CALL PRT_R ;
    END ;
ELSE IF SUBSTR(INREC,7,6) = '068202' &
    SUBSTR(INREC,74,1) < '5' THEN DO;
    SUBSTR(INREC,74,1) = 'D' ;
    CALL PRT_R ;
    END ;
ELSE IF SUBSTR(INREC,7,6) = '068203' &
    SUBSTR(INREC,74,1) < '5' THEN DO;
    SUBSTR(INREC,74,1) = 'D' ;
    CALL PRT_R ;
    END ;
WRITE FILE(TAPEOUT) FROM(INREC);
WCN = WCN + 1 ;
GO TO BEGIN;
PRT_R: PROC ;
    PUT SKIP EDIT(INREC) (A(100)) ;
END PRT_R ;
COV_R : PROC ;
    TTT='';
    IF SUBSTR(TT,19,2)='00' THEN
    DO ;
        CALL HGBATCH(FUNC,RECODE,ADDR,LA,DDD,LB) ;
        OADDR = DDD;
        TAA = TA , BY NAME ;
        DO I=1 TO 572;
            IF TA.JOSA=DJOSA(I) THEN HJ = HJCHK(I) ;
        END;
    ELSE DO;
        CALL HGBATCH(FUNC,RECODE,NAME,LC,ABC,LD) ;
        ONAME = ABC ;
        TBB = TB , BY NAME ;
    END;
    IF RECODE '= ' THEN DO ;
        PUT SKIP(2) EDIT('HAW ERROR : ',RECODE,STRING(TT))
            (A,X(2)) ;
        ECNT = ECNT + 1 ;
        GOTO RD ;
    END ;
    OCNT = OCNT + 1 ;
END COV_R ;
CLOS: CLOSE FILE(DISKIN),FILE(TAPEOUT);
    PUT SKIP(5) EDIT(' IN CNT = ',ICNT)(A,F(6));
    PUT SKIP(5) EDIT(' OUT CNT = ',OCNT)(A,F(6));
    PUT SKIP(5) EDIT(' INPUT CNT = ',ICN)(A,F(6));
    PUT SKIP(5) EDIT(' BACK-UP CNT = ',WCN)(A,F(6));
END RIVER;
/*
//GO.SYSIN DD UNIT=DISK,DCB=(RECFM=FB,BLKSIZE=4000,LRECL=100),
// DSN=A33.VH.ACT,DISP=(OLD,KEEP),VOL=SER=SFS803
//GO.DISKIN DD DSN=A33.VH.MS,DISP=SHR
//GO.TAPEOUT DD UNIT=DISK,DCB=(RECFM=FB,BLKSIZE=20000,LRECL=100),
// DSN=A33.VH.VSAM,DISP=(OLD,KEEP),VOL=SER=BOSS1K
/** SPACE=(TRK,(1500,100))

```

```

00019000
00019100
00019200
00019300
00019400
00019500
00019500
00019700
00019800
00019900
00020000
00020100
00020200
00020300
00020400
00020500
00020600
00020700
00020800
00020900
00021000
00021100
00021200
00021300
00021400
00021500
00021600
00021700
00021800
00021900
00022000
00022100
00022200
00022300
00022400
00022500
00022600
00022700
00022800
00022900
00023000
00023100
00023200
00023300
00023400
00023500
00023600
00023700
00023800
00023900
00024000
00024100
00024200
00024300
00024400
00024500
00024600
00024700
00024800
00024900
00025000
00025100
00025200

```

A33,86.147,11:40:02,A33.PROGRAM.LIBRARY(MSORT)

```
//A33VHUST JOB CLASS=A,MSGCLASS=X,NOTIFY=A33 00000100
//STEP1 EXEC PGM=MSORT,PARM='SIZE(MAX)' 00000200
//SYSOUT DD SYSOUT=* 00000300
//SYSPRINT DD SYSOUT=* 00000400
//SORTIN DD DSN=A33.VH.VSAM,UNIT=DISK,DISP=(OLD,KEEP), 00000500
// DCB=(RECFM=FB,LRECL=100,BLKSIZE=20000), 00000600
// VOL=SER=BOSSSTR 00000700
//SORTOUT DD DSN=A33.VH.LDY,UNIT=DISK,DISP=(NEW,KEEP), 00000800
// DCB=(RECFM=FB,LRECL=100,BLKSIZE=4000), 00000900
// VOL=SER=BOSSSTR,SPACE=(TRK,(1500,50)) 00001000
//SORTWK01 DD UNIT=DISK,SPACE=(CYL,150),VOL=SER=SORTWK 00001100
//SYSIN DD * 00001200
SORT FIELDS=(7,12,A,19,2,A),FORMAT=BI,FILSZ=E120000 00001300
/* 00001400
// 00001500
```

A33,86.055,18:46:51,A33.PROGRAM.LIBRARY(MULT)

```
//A33VHWC JOB CLASS=A,MSGCLASS=X,NOTIFY=A33 00000100
//JOB CAT DD DSN=CATALOG.VBOS005,DISP=SHR 00000200
//STEP1 EXEC PGM=IDCAMS,REGION=512K 00000300
//DD1 DD VOL=SER=BOS005,UNIT=DISK,DISP=OLD 00000400
//SYSPRINT DD SYSOUT=* 00000500
//SYSIN DD * 00000600
DELETE A33.VH.MS 00000700
DEFINE CLUSTER - 00000800
(NAME(A33.VH.MS) - 00000900
VOLUME(BOS005) - 00001000
INDEXED - 00001100
FILE(DD1) - 00001200
RECORDS(200000 100) - 00001300
KEYS(14 6) - 00001400
FREESPACE(15 15) - 00001500
RECORDSIZE(100 100)) - 00001600
CATALOG(CATALOG.VBOS005) 00001700
/* 00001800
//STEP2 EXEC PLIFCLG,REGION=2048K 00001900
//PLI.SYSIN DD * 00002000
* PROCESS GS,NEST,A,XREF,OPT(2); 00002100
CREATE PROC OPTIONS(MAIN); 00002200
DCL TPI FILE RECORD INPUT; 00002300
DCL TPO FILE RECORD KEYED SEQUENTIAL OUTPUT ENV(VSAM); 00002400
DCL IN CHAR(100); 00002500
DCL OCNT PIC '(6)9' INIT(0); 00002600
DCL ONCODE BUILTIN; 00002700
ON KEY(TPO) BEGIN; 00002800
IF ONCODE=52 THEN DO; 00002900
PUT SKIP EDIT(TKEY) (A); 00003000
GO TO RD; END; 00003100
END; 00003200
ON ENDFILE(TPI) GO TO LAST; 00003300
OPEN FILE(TPI),FILE(TPO); 00003400
RD: READ FILE(TPI) INTO(IN); 00003500
WRITE FILE(TPO) FROM(IN) KEYFROM(SUBSTR(IN,7,14)); 00003600
OCNT = OCNT + 1; 00003700
GOTO RD; 00003800
LAST:CLOSE FILE(TPI),FILE(TPO); 00003900
PUT SKIP(2) EDIT('OUTPUT COUNT',OCNT)(A,F(10)); 00004000
END CREATE; 00004100
/* 00004200
//GO.SYSOUT DD SYSOUT=* 00004300
//GO.TPI DD DSN=A33VHTT,DCB=(RECFM=FB,BLKSIZE=20000,LRECL=100), 00004400
// UNIT=TAPE,VOL=SER=TN3407,DISP=(OLD,KEEP),LABEL=(2,BLP) 00004500
//GO.TPO DD DSN=A33.VH.MS,DISP=SHR 00004600
// 00004700
```

A33,86.055,17:46:50,A33.PROGRAM.LIBRARY(TDELETE)

```
//A33VHTDL JOB CLASS=A,MSGCLASS=X,NOTIFY=A33,USER=A33,PASSWORD=PINE 00000100
//JOB CAT DD DSN=CATALOG.VBOS005,DISP=SHR 00000200
//DELETE EXEC PLIFCLG 00000300
//PLI.SYSIN DD * 00000400
* PROCESS S,GS,NEST; 00000500
KANG: PROC OPTIONS(MAIN); 00000600
  DCL TPP FILE RECORD INPUT; 00000700
  DCL TPI FILE RECORD ENV(VSAM); 00000800
  DCL TA CHAR(80); 00000900
  DCL TB CHAR(100); 00001000
  DCL DKEY CHAR(6); 00001100
  DCL DCNT PIC '99999' INIT(0); 00001200
  ON ENDFILE(TPP) GOTO LAST; 00001300
  OPEN FILE(TPP),FILE(TPI) SEQUENTIAL UPDATE; 00001400
  RD: READ FILE(TPP) INTO(TA); 00001500
  IF SUBSTR(TA,13,1) = 'D' THEN GOTO RD; 00001600
  DKEY = SUBSTR(TA,7,6); 00001700
  RD1: READ FILE(TPI) INTO(TB); 00001800
  RDA: IF DKEY = SUBSTR(TB,7,6) THEN 00001900
  DO ; 00002000
    SUBSTR(TB,74,1)='D' ; 00002100
    REWRITE FILE(TPI) FROM(TB); 00002200
    DCNT = DCNT + 1 ; 00002300
    GO TO RD1; 00002400
  END ; 00002500
  ELSE IF DKEY < SUBSTR(TB,7,6) THEN 00002600
  DO ; 00002700
    RDB: READ FILE(TPP) INTO(TA) ; 00002800
    IF SUBSTR(TA,13,1) = 'D' THEN GOTO RDB ; 00002900
    DKEY = SUBSTR(TA,7,6) ; 00003000
    GO TO RDA; 00003100
  END ; 00003200
  ELSE GOTO RD1 ; /* DKEY > SUBSTR(TB,7,6) */ 00003300
  LAST: CLOSE FILE(TPP),FILE(TPI) ; 00003400
  PUT SKIP EDIT('DELETE GAGU CNT =',DCNT)(A,F(5)); 00003500
END KANG ; 00003600
/* 00003700
//GO.SYSPRINT DD SYSOUT=* 00003800
//GO.SYSOUT DD SYSOUT=* 00003900
//GO.TPP DD DSN=A33.PROGRAM.LIBRARY(DATAN),DISP=(OLD,KEEP) 00004000
//GO.TPI DD DSN=A33.VH.MS,DISP=SHR 00004100
// 00004200
```

A33,86.055,17:46:41,A33.PROGRAM.LIBRARY(DELETE)

```
//A33VHDEL JOB CLASS=A,MSGCLASS=X,NOTIFY=A33          00000100
//DELETE EXEC PLIFCLG                                00000200
//PLI.SYSIN DD *                                     00000300
* PROCESS S,GS,NEST;                                00000400
KANG: PROC OPTIONS(MAIN);                             00000500
  DCL TPP FILE RECORD INPUT;                          00000600
  DCL TPI FILE RECORD ENV(VSAM);                      00000700
  DCL TA      CHAR(80);                               00000800
  DCL TB      CHAR(100);                             00000900
  DCL DKEY    CHAR(6);                               00001000
  DCL (ICNT,DCNT) PIC '99999' INIT(0) ;             00001100
  ON ENDFILE(TPP) GOTO LAST ;                         00001200
  OPEN FILE(TPP),FILE(TPI) SEQUENTIAL UPDATE;       00001300
  RD:  READ FILE(TPP) INTO(TA);                      00001400
      IF SUBSTR(TA,13,1) = 'D' THEN GOTO RD;         00001500
      ICNT = ICNT + 1;                               00001600
      DKEY = SUBSTR(TA,7,6);                          00001700
  RD1: READ FILE(TPI) INTO(TB);                      00001800
  RDA: IF DKEY = SUBSTR(TB,7,6) THEN DO ;            00001900
      DELETE FILE(TPI);                              00002000
      DCNT = DCNT + 1 ;                             00002100
      GO TO RD1;                                     00002200
  END ;                                              00002300
  ELSE IF DKEY < SUBSTR(TB,7,6) THEN DO ;           00002400
      READ FILE(TPP) INTO(TA) ;                     00002500
      GO TO RDA;                                    00002600
  END ;                                              00002700
  ELSE GOTO RD1 ;                                   00002800
  LAST: CLOSE FILE(TPP),FILE(TPI) ;                 00002900
        PUT SKIP EDIT(' INPUT GAGU CNT =',ICNT)(A,F(5)); 00003000
        PUT SKIP EDIT('DELETE GAGU CNT =',DCNT)(A,F(5)); 00003100
  END KANG ;                                         00003200
/*                                                  00003300
//GO.SYSPRINT DD SYSOUT=*                          00003400
//GO.SYSOUT DD SYSOUT=*                            00003500
//GO.TPP DD DSN=A33.PROGRAM.LIBRARY(DATAN),DISP=SHR 00003600
//GO.TPI DD DSN=S93.VH.MS,DISP=SHR                00003700
//                                                  00003800
```

A33,86.055,17:40:15,A33.PROGRAM.LIBRARY(SVHWDI93)

```
//A33VHWDI JOB CLASS=A,MSGCLASS=X,NOTIFY=A33 00000100
//STEP1 EXEC PLIFCLG,REGION=1024K 00000200
//PLI.SYSIN DD * 00000300
* PROCESS S,GS,A,XREF,OPT(2); 00000400
LEE: PROC OPTIONS(MAIN); 00000500
DCL TPI FILE RECORD INPUT ; 00000600
DCL TPO FILE RECORD OUTPUT ; 00000700
DCL INR CHAR(100); 00000800
DCL I IN1 DEF INR, 00000900
      2 BB CHAR(6), 00001000
      2 GI CHAR(14), 00001100
      2 DATA CHAR(80); 00001200
DCL MM CHAR(100); 00001300
'DCL I PIC'9'; 00001400
ON ENDFILE(TPI) GO TO LAST; 00001500
ON ENDFILE(SYSIN) GO TO PASS; 00001600
OPEN FILE(TPI), FILE(TPO), FILE(SYSPRINT) LINESIZE(132); 00001700
RD: GET SKIP EDIT(MM)(A(100)); 00001800
SUBSTR(INR,1,100) = MM; 00001900
WRITE FILE(TPO) FROM(INR); 00002000
GO TO RD ; 00002100
PASS: READ FILE(TPI) INTO(INR); 00002200
IF BB = '101414' & SUBSTR(GI,1,12) = '001404010101' 00002300
  THEN SUBSTR(GI,1,12) = '001404090101' ; 00002400
IF GI = '00750704010104' & SUBSTR(DATA,1,7) = 'DLRMAWK' 00002500
  THEN GI = '00750704010111' ; 00002600
IF GI = '03010307020300' & SUBSTR(DATA,1,7) = 'DBSRHRWL' 00002700
  THEN GI = '03010307020303' ; 00002800
IF BB = '159359' & SUBSTR(GI,1,12) = '059309080101' 00002900
  THEN SUBSTR(GI,1,12) = '059309080201' ; 00003000
IF BB = '167663' & GI = '06751509010101' 00003100
  THEN GI = '06761509010101' ; 00003200
IF BB = '171022' & GI = '07100711010101' & 00003300
  SUBSTR(DATA,1,9) = 'JQORTNWJD' 00003400
  THEN GI = '07100711010111' ; 00003500
IF BB = '174717' & GI = '07470704010102' & 00003600
  SUBSTR(DATA,1,9) = 'RKDQUDTJS' 00003700
  THEN GI = '07470704010109' ; 00003800
IF BB = '178435' & SUBSTR(GI,1,12) = '078402080101' 00003900
  THEN SUBSTR(GI,1,12) = '078402080201' ; 00004000
IF GI = '07910513010105' & 00004100
  SUBSTR(DATA,1,7) = 'WJDVHCS' 00004200
  THEN GI = '07910513010106' ; 00004300
IF BB = '186512' & SUBSTR(GI,1,12) = '086506030101' 00004400
  THEN SUBSTR(GI,1,12) = '086506020101' ; 00004500
IF GI = '08780607010104' & 00004600
  SUBSTR(DATA,1,9) = 'RLAWNNDNJS' 00004700
  THEN GI = '08780607010103' ; 00004800
WRT: WRITE FILE(TPO) FROM(INR); 00004900
GO TO PASS; 00005000
/*UPR: DO I = 1 TO 4; 00005100
      IF CH(I) = (5) THEN GO TO CHR; 00005200
      END; 00005300
CHR: CH(I) = '8412'; 00005400
WRITE FILE(TPO) FROM(INR); 00005500
GO TO PASS; */ 00005600
LAST: CLOSE FILE(TPI),FILE(TPO); 00005700
END LEE; 00005800
/* 00005900
//GO.SYSIN DD * 00006000
/* 00006100
//GO.TPI DD DSN=SVHT393,UNIT=DISK,VOL=SER=SORTWK, 00006200
// DISP=(OLD,KEEP),DCB=(RECFM=FB,BLKSIZE=20000,LRECL=100) 00006300
//
```

```
//GO.TPO DD DSN=SVHT293,UNIT=DISK,VOL=SER=BOS003, 00006400
// DISP=(OLD,KEEP),DCB=(RECFM=FB,BLKSIZE=20000,LRECL=100) 00006500
//GO.SYS$PRINT DD SYSOUT=* 00006600
//GO.SYSOUT DD SYSOUT=* 00006700
// 00006800
```



A33,86.055,18:46:29,A33.PROGRAM.LIBRARY(MULTY)

```
//A33VHCRT JOB CLASS=A,MSGCLASS=X,NOTIFY=A33 00000100
//JOB CAT DD DSN=CATALOG.VBOS005,DISP=SHR 00000200
//STEP1 EXEC PGM=IDCAMS,REGION=1024K 00000300
//SYSPRINT DD SYSOUT=* 00000400
//SYSOUT DD SYSOUT=* 00000500
//DD1 DD VOL=SER=BOS005,UNIT=DISK,DISP=OLD 00000600
DELETE A33.VH.MS 00000700
DEFINE CLUSTER - 00000800
(NAME(A33.VH.MS) - 00000900
VOLUME(BOS005) - 00001000
INDEXED - 00001100
,SHAREOPTIONS(2 3) - 00001200
RECORDS(200000 100) - 00001300
KEYS(14 6) - 00001400
FREESPACE(10 10) - 00001500
RECORDSIZE(100 100)) - 00001600
CATALOG(CATALOG.VBOS005) 00001700
//STEP2 EXEC PLIFCLG,REGION=2048K 00001800
//PLI.SYSIN DD * 00001900
* PROCESS S,GS ; 00002000
CREATE PROCEDURE OPTIONS(MAIN) ; 00002100
DCL TPI FILE RECORD INPUT ; 00002200
DCL 1 TT CHAR(100); 00002300
DCL 1 TA DEF TT, 00002400
2 CARD CHAR(6), 00002500
2 JOSA PIC '(4)9', 00002600
2 KEYA CHAR(10), 00002700
2 GU CHAR(6), 00002800
2 ADDR CHAR(41), 00002900
2 FILA CHAR(33); 00003000
DCL 1 TB DEF TT, 00003100
2 CARD1 CHAR(6), 00003200
2 JOSA1 PIC '(4)9', 00003300
2 KEYA1 CHAR(10), 00003400
2 NAME CHAR(20), 00003500
2 NUM CHAR(33), 00003600
2 BLK CHAR(27); 00003700
DCL TPO FILE RECORD KEYED SEQUENTIAL OUTPUT ENV(VSAM); 00003800
DCL 1 TTT CHAR(100); 00003900
DCL 1 TAA DEF TTT, 00004000
2 CARD CHAR(6), 00004100
2 JOSA PIC '(4)9', 00004200
2 KEYA CHAR(10), 00004300
2 GU CHAR(6), 00004400
2 HJ PIC '(6)9', 00004500
2 OADDR CHAR(41), 00004600
2 FIL1 CHAR(27); 00004700
DCL 1 TBB DEF TTT, 00004800
2 CARD1 CHAR(6), 00004900
2 JOSA1 PIC '(4)9', 00005000
2 KEYA1 CHAR(10), 00005100
2 ONAME CHAR(20), 00005200
2 NUM CHAR(33), 00005300
2 BLK CHAR(27); 00005400
DCL HGBATCH ENTRY OPTIONS(ASSEMBLER) ; 00005500
DCL LA FIXED BIN(15) INIT(40); 00005600
DCL LB FIXED BIN(15) INIT(40); 00005700
DCL LC FIXED BIN(15) INIT(16); 00005800
DCL LD FIXED BIN(15) INIT(10); 00005900
DCL FUNC CHAR(2) INIT('IF') ; 00006000
DCL ABC CHAR(20) INIT('') ; 00006100
DCL DDD CHAR(40) INIT('') ; 00006200
DCL RECODE CHAR(2) INIT(' ') ; 00006300
```

```

DCL (OCNT,ICNT,ECNT) PIC '(6)9' INIT(0) ; 00006400
DCL DJOSA(662) PIC '(4)9' INIT 00006500
(0001,0002,0003,0005,0006,0007,0009,0010,1011,2011,0013,0014, 00006600
0015,0017,0018,0019,0021,0022,0023,0025,0026,0027,0029,0030, 00006700
0031,0033,0034,0035,0037,0038,0039,0041,0042,0043,0045,0046, 00006800
0047,0049,0050,0051,0053,0054,0055,0057,0058,0059,0061,0062, 00006900
0063,0065, 00007000
0066,0067,0069,0070,0071,0073,0074,0075,0077,0078, 00007100
0079,0081,0082,0083,0085,0086,0087,0089,0090,0091, 00007200
0093,0094,0095,0097,0098,0099,0101,0102,0103,0105, 00007300
0106,0107,0109,0110,0111,0113,0114,0115,0117,0118, 00007400
0119,0121,0122,0123,0125,0126,0127,0129,0130,0131, 00007500
0133,0134,0135,0137,0139,0141,0142,0143,0145,0147, 00007600
0149,0150,0151,0153,0154,0155,0157,0158,0159,0161, 00007700
0162,1163,2163,0165,0166,0167,0169,0170,0171,0173, 00007800
0174,0175,0177,0178,0179,0181,0182,0183,0185,0186, 00007900
0187,0189,0190,0191,0193,0194,0195,0197,0198,0199, 00008000
0201,0202,0203,0205,0206,0209,0210,0211,0213,0214, 00008100
0215,0217,0218,0219,0221,0222,0223,0225,0226,0227, 00008200
0229,0230,0250,0251,0253,0254,0255,0257,0258,0259, 00008300
0261,0262,0263,0265,0266,0267,0269,0270,0271,0273, 00008400
0274,0275,0277,0278,0279,0281,0282,0283,0285,0286, 00008500
0287,0289,0290,0291,0293,0294,0295,0297,0298,0299, 00008600
0301,0302,1302,2302,3302, 00008700
0303,0305,0306,0307,0309,0310,0311,0313, 00008800
0314,0315,0317,0318,0319,0321,0322,0323,0325,0326, 00008900
0327,0329,0330,0331,0333,0334,0335,0337,0338,0351,0353, 00009000
0354,0355,0357,0358,0359,0361,0362,0363,0365,0366, 00009100
0367,0369,0370,0371,0373,0374,0375,0377,0378,0379, 00009200
0381,0382,0383,0385,0386,0387,0389,0390,0391,0393, 00009300
0394,0395,0397,0398,0399,0401,0403,0405,0406,0407, 00009400
0409,0410,0411,0413,0414,0415,0417,0418,0419,0421, 00009500
0422,0423,0425,0426,0427,0429,0430,0431,0433,0434, 00009600
0502,0503,0505,0506,0507,0509,0510,0511,0513,0514, 00009700
0515,0517,0518,0519,0521,0522,0523,0525,0526,0527, 00009800
0529,0530,0531,0533,0534,1535,2535,3535,4535,0537, 00009900
0538,1539,2539,0541,0542,0543,0545,0546,0547,0549,0550, 00010000
0551,0553,0554,0555,0557,0558,0559,0561,0562,0563, 00010100
0565,0566,0567,0571,0572,0573,1574,2574,0575,0576,0577, 00010200
0578,0579,0580,0581,0582,1583,2583,3583,0584,0589, 00010300
0590,0591,0592,0593,0594,0595,0603,0605,0606,0607, 00010400
1607,2607,3607, 00010500
0609,0610,0611,0613,0614,0618,0619,0621,0622,0623,0625, 00010600
0626,0627,0629,0630,0631,0633,0634,0635,0671,0672, 00010700
0673,0674,0675,0676,0680,0681,0682,0683,0684,0685, 00010800
0686,0687,0688,0689,0690,0691,0692,0701,0702,0703, 00010900
0705,0706,0707,0709,0710,1711,2711,3711,0713,0714, 00011000
0715,0717,0718,0719,0725,0726,0727,0729,0730,0731, 00011100
0733,0734,0735,0737,0738,0739,0741,0742,0743,0745, 00011200
0746,0747,0749,0750,0751,0753,0754,0755,0771,0772, 00011300
0773,0774,0775,0776,0777,0778,0779,0783,0784,0785, 00011400
0786,0787,0788,0789,0790,0791,0792,0793,0794,0795, 00011500
0796,1803,2803,0805,0806,0807,0809,0810,0811,0813, 00011600
0814,0815,0817,0818,0819,0821,0822,0827,0829,0830, 00011700
0831,0833,0834,0835,0837,0838,0839,0841,0842,0843, 00011800
0845,0846,0847,0849,0850,0851,0853,0854,0855,0856, 00011900
0859,0861,0862,0863,0865,0866,0867,0869,0871,0872, 00012000
0873,0874,0875,0876,0877,0878,0879,0880,0881,0882, 00012100
0883,0884,0885,0886,0890,0891,0892,0893,0894,0895, 00012200
0896,0897,0898,0899,0981,0982, 00012300
0903,0905,0906,0907,0992,0993) ; 00012400
DCL HJCHK(662) PIC '(6)9' INIT( 00012500
111735,112630,112631,111540,112717,111542,111551,112636,112123,112123, 00012600

```

111223,112611,112534,111639,112025,111417,111412,111513,112028,111631, 00012700  
111919,111513,111616,112027,111722,111529,112324,111814,112030,111820, 00012800  
112134,111512,112234,111630,111917,112319,112024,112523,112020,112528, 00012900  
111116,111716,111543,111222,111919,112331,111515,111728,111228,111627, 00013000  
112518,111532,111537,112020,112517,112120,111114,111616,111813,112031, 00013100  
111428,111546,111536,111421,111514,111723,112034,111128,112422,112235, 00013200  
111227,111437,111120,111550,111228,111728,112212,112525,111823,111629, 00013300  
112119,111711,112414,111811,112319,111548,111428,112028,111522,112323, 00013400  
111520,111416,111412,112517,112222,111548,112526,111930,111540,111926, 00013500  
112635,112411,112725,111630,111636,112022,111620,112613,112528,112514, 00013600  
111217,111927,111621,112117,111124,112530,112425,111325,111439,111627, 00013700  
111635,112619,112619,112525,111739,111613,111741,112011,111515,111226, 00013800  
112123,111745,111734,112114,112715,112626,112612,111415,111632,112720, 00013900  
112629,112314,111121,112714,112725,112615,112714,112616,112626,112626, 00014000  
112625,112616,111631,112217,111433,112320,111222,111318,111518,112416, 00014100  
112323,112226,111726,112318,111631,111219,111427,112233,111815,112331, 00014200  
112314,112718,211227,211827,211228,211828,211828,211818,211424,212021, 00014300  
211511,211425,211538,211221,211630,211523,211114,212013,211619,211526, 00014400  
211615,212021,211615,211727,211626,211641,211226,211537,211329,211637, 00014500  
211732,211321,211644,211623,212017,211617,211516,211411,211614,211822, 00014600  
211617,211651,211651,211651,211651,211739,211619,211619, 00014700  
211535,211629,211733,211826,211832,211725,211637,211712,211636,211922, 00014800  
211223,212013,211533,211617,211819,211424,211415,211624,211526,211718, 00014900  
211817,211716,211424,212018, 00015000  
221628,221235,221222,221325,221225,221324,221331, 00015100  
221320,221412,221215,221514,221211,221318,221217,221128,221523,221521, 00015200  
221611,221214,221126,221319,221328,221523,221411,221326,221129,221217, 00015300  
221215,221331,221315,221516,221417,221516,221323,221211,221334,221521, 00015400  
221514,231219,231423,231432,231319,231415,231220,231115,231119,231113, 00015500  
231318,231313,231313,231328,231413,231322,231417,231219,231212,231326, 00015600  
231427,231318,231424,231419,231316,311712,311714,311114,311127,311120, 00015700  
311115,311126,311237,311218,311239,311218,311213,311211, 00015800  
311227,311220,311226,311320,311319,311313,311424,311413,311413,311414, 00015900  
311412,311424,311520,311520,311520,311520,311511,311518,311513,311513, 00016000  
311615,311614,311611,311620,311411,311423, 00016100  
321127,321117,321111,321219,321220, 00016200  
321223,321416,321420,321525,321417,321321,321322,321314,321612,321122, 00016300  
313412, 00016400  
313912,313712,314511,314511,313411,314536,314434,314737,313537,314633, 00016500  
314137,314339,315033,315033,315033,313631,324411,323712,323311,323732, 00016600  
323133,323432,321622,331128,331116,331118,331125,331125,331125,331125, 00016700  
331218,331213,331314, 00016800  
331313,331121, 00016900  
340115,340222,340112,340129,340126,340214,340117,340224,340231, 00017000  
340212,340231, 00017100  
340245,341120,341112,333411,333931,333831,333636,334034,333138,344112, 00017200  
343211,343111,343811,343536,344331,343736,344541,344433,344242,344032, 00017300  
343137,343432,351141,351135,351127,351120,351117,351123,351126,351418, 00017400  
351226,351226,351226,351412,351511,351217,351415,351312,351216,360227, 00017500  
360115,360211,360122,260317,360112,360332,360220,360213,360119,360121, 00017600  
360314,360119, 00017700  
361122,361234,361316,361225,361213,361112,353411,353112,354334,353633, 00017800  
354234,354035,353135,353841,354136,364011,364811,363211,364211,363731, 00017900  
363938,364934,364133,364531,363134,363536,364641,364432,364042,371121, 00018000  
371121,371134,371114,371123,371111,371223,371228,371211,371319,371519, 00018100  
371320,371314,371423,371414,371420,381111,381119,381117,381120,381135, 00018200  
381134,381116,381516,381143,381128,381142,381242,381230,381227,381213, 00018300  
381240,381225,381216,381215,381818,381820,381423,381331,381325,381323, 00018400  
381327,381612,381618,381617,381423,381427,374913,374811,373411,375311, 00018500  
373912,374332,374734,374432,373343,374536,373731,373440,374034,374634, 00018600  
375131,374032,383412,383612,383735,384431,384834,383334,384737,384242, 00018700  
384336,383836,383146,384537,391119,391125,391114,391116,393111,393131 00018800  
) , 00018900

DCL ONCODE BUILTIN ;	00019000
ON KEY(TPO) BEGIN ;	00019100
IF ONCODE = 52 THEN DO ;	00019200
PUT SKIP EDIT ('DATA DUPLICATED', 'KEY=', SUBSTR(TT, 7, 14), TT)	00019300
(A, A, A, X(2), A(100));	00019400
GO TO RD;                    END ;	00019500
END,	00019600
ON ENDFILE(TPI) GO TO LAST ;	00019700
OPEN FILE(TPI), FILE(TPO) ;	00019800
RD:  READ FILE(TPI) INTO(TT) ;	00019900
ICNT = ICNT+1 ;	00020000
TTT='';	00020100
IF SUBSTR(TT, 19, 2)='00' THEN	00020200
DO ;	00020300
CALL HGBATCH(FUNC, RECODE, ADDR, LA, DDD, LB) ;	00020400
OADDR = DDD;	00020500
TAA = TA , BY NAME ;	00020600
DO I=1 TO 562;	00020700
IF TA.JOSA=DJOSA(I) THEN HJ = HJCHK(I) ;	00020800
END;	00020900
END ;	00021000
ELSE DO;	00021100
CALL HGBATCH(FUNC, RECODE, NAME, LC, ABC, LD) ;	00021200
ONAME = ABC ;	00021300
TBB = TB , BY NAME ;	00021400
END;	00021500
IF RECODE '=' THEN DO ;	00021600
PUT SKIP(2) EDIT('HAR ERROR : ', RECODE, STRING(TT))	00021700
(A, X(2)) ;	00021800
ECNT = ECNT + 1 ;	00021900
GOTO RD ;	00022000
END ;	00022100
OCNT = OCNT + 1 ;	00022200
WRITE FILE(TPO) FROM(TTT) KEYFROM(SUBSTR(TTT, 7, 14)) ;	00022300
GOTO RD ;	00022400
LAST:CLOSE FILE(TPI), FILE(TPO) ;	00022500
PUT SKIP(2) EDIT(ICNT, ECNT, OCNT)(3 F(10)) ;	00022600
END CREAT;	00022700
/*	00022800
//GO.TPI DD DSN=A33.VH.T1, DCB=(RECFM=FB, BLKSIZE=20000, LRECL=100),	00022900
//          UNIT=DISK, VOL=SER=BQS003, DISP=(OLD, KEEP)	00023000
//GO.TPO DD DSN=A33.VH.MS, DISP=SHR	00023100
//	00023200

A33.86.055,18:47:18,A33.PROGRAM.LIBRARY(VSAMDIR)

```
//A33VHDIR JOB CLASS=A,MSGCLASS=X,NOTIFY=A33 00000100
//JOB CAT DD DSN=CATALOG.VBOS005,DISP=SHR 00000200
//STEP1 EXEC PLIFCLG,REGION=3072K 00000300
//PLI.SYSIN DD * 00000400
* PROCESS S,GS,NEST,A,A,XREF,OPT(2); 00000500
BABO : PROCEDURE OPTIONS(MAIN); 00000600
  DCL TPP FILE RECORD INPUT , 00000700
    TB CHAR(100); 00000800
  DCL TPI FILE RECORD KEYED ENV(VSAM), 00000900
    TA CHAR(100); 00001000
  DCL (ONCODE,SUBSTR) BUILTIN; 00001100
  DCL (CNT1,CNT2,CNT3) PIC '9999' INIT(0); 00001200
  DCL (I,J,K,L,M) PIC '9999' INIT(0); 00001300
  DCL (SKEY) CHAR(14) INIT(' '); 00001400
  OPEN FILE(TPP), FILE(TPI) DIRECT UPDATE, 00001500
    FILE(SYSPRINT) LINESIZE(132); 00001600
  ON KEY(TPI) BEGIN ; 00001700
    IF ONCODE = 52 THEN 00001800
      DO; PUT SKIP(1) EDIT ('DUP 1 ',TB) (A(12),A); 00001900
        GOTO RD; 00002000
      END; 00002100
    IF ONCODE = 51 00002200
      THEN PUT SKIP(1) EDIT ('NOT-FOUND ',TB) (A(12),A); 00002300
      GOTO RD; 00002400
    END; 00002500
  ON ENDFILE(TPP) GOTO LAST ; 00002600
  RD : READ FILE(TPP) INTO(TB); 00002700
    SKEY = SUBSTR(TA,1,14); 00002800
    READ FILE(TPI) INTO(TA) KEY(SKEY); 00002900
    PUT SKIP EDIT(TA) (A(100)); 00003000
    DELETE FILE(TPI) KEY(SKEY); 00003100
  GOTO RD ; 00003200
  LAST : CLOSE FILE(TPP),FILE(TPI) ; 00003300
  END BABO; 00003400
/* 00003500
//GO.TPP DD DSN=S93.VH.ACT,DISP=(OLD,KEEP),UNIT=DISK,VOL=SER=SPS803, 00003600
// DCB=(RECFM=FB,BLKSIZE=4000,LRECL=100) 00003700
//GO.TPI DD DSN=A33.VH.MS,DISP=SHR 00003800
//GO.SYSPRINT DD SYSOUT=* 00003900
// 00004000
```

A33,86.055,18:47:35,A33.PROGRAM.LIBRARY(VSAMSEQ)

```
//A33VHTAB JOB CLASS=A,MSGCLASS=X,NOTIFY=A33 00000100
//JOB CAT DD DSN=CATALOG.VBOS005,DISP=SHR 00000200
//STEP1 EXEC PLIFCLG,REGION=3072K 00000300
//PLI.SYSIN DD * 00000400
* PROCESS S,GS,NEST,A,A,XREF,OPT(2); 00000500
BABO : PROCEDURE OPTIONS(MAIN); 00000600
  DCL TPI FILE RECORD KEYED ENV(VSAM), 00000700
  TA CHAR(100); 00000800
  DCL (ONCODE,SUBSTR) BUILTIN; 00000900
  DCL (DCNT,CNT2,CNT3) PIC '9999' INIT(0) ; 00001000
  DCL (I,J,K,L,M) PIC '9999' INIT(0) ; 00001100
  DCL (KEYA,KEYB) CHAR(14) INIT(' '); 00001200
  OPEN FILE(TPI) SEQUENTIAL UPDATE, 00001300
  FILE(SYSPRINT) LINESIZE(132); 00001400
  ON KEY(TPI) BEGIN ; 00001500
  IF ONCODE = 52 THEN 00001600
    DO; PUT SKIP(1) EDIT ('DUP 1 ',TB) (A(12),A); 00001700
      GOTO RD; 00001800
    END; 00001900
  IF ONCODE = 51 00002000
    THEN PUT SKIP(1) EDIT ('NOT-FOUND ',TB) (A(12),A); 00002100
    GOTO RD; 00002200
  END; 00002300
  ON ENDFILE(TPI) GOTO LAST ; 00002400
  RD : READ FILE(TPI) INTO(TA); 00002500
  IF SUBSTR(TA,7,6) = '030306' THEN DO ; 00002600
    DCNT = DCNT + 1 ; 00002700
    PUT SKIP EDIT(TA)(A(100)); 00002800
    DELETE FILE(TPI) ; END ; 00002900
  GOTO RD ; 00003000
  LAST : CLOSE FILE(TPI) ; 00003100
    PUT SKIP EDIT('DELETE CNT = ',DCNT)(A,F(4)); 00003200
  END BABO; 00003300
/* 00003400
//GO.TPI DD DSN=A33.VH.MS,DISP=SHR 00003500
//GO.SYSPRINT DD SYSOUT=* 00003600
// 00003700
```

## 7. 결 과 표 양 식

< 표 1 >      조사구별 인구수 및 가구수 (매월)

조사구 번 호	인 구 수			유배우 여 자 (15~49세)	가구수	전 입	전 출	출 생	사 망
	계	남자	여자						



< 표 2 >

### 연령 및 배우관계별 인구수

(전국, 대도시지부, 군부)

연령 (Aged)	총 수 (Total)			이 혼 (Single)			유 배우 (Married)			이 혼 (Divorced)			사 별 (Widowed)		
	계 B.S	남 Male	여 Fe- Male	계 B.S	남 Male	여 Fe- Male	계 B.S	남 Male	여 Fe- Male	계 B.S	남 Male	여 Fe- Male	계 B.S	남 Male	여 Fe- Male
0~4															
0															
1															
2															
3															
4															
5~9															
10~14															
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															
77															
78															
79															
80															
81															
82															
83															
84															
85+															

- 집계요령 : 1) 전국, 대도시, 시부, 군부  
 2) 가구표에서 집계  
 3) 분기별 집계

〈 표 3 〉 교육정도별 인구수(Pop by educational attainment)

(전국, 시부, 군부)

연령 (Aged)	총 수 (Total)			불 취 학 (Never attended)			국민학교 (Primary school)			중 학교 Middle school			고등학교 (High school)			대 학 이 상 (College or more)		
0~4																		
0																		
1																		
2																		
3																		
4																		
5~9																		
⋮																		
10~14																		
⋮																		
⋮																		
85+																		

집계요령 : 1) 전국, 시부, 군부  
 2) 가구표에서 집계  
 3) 분기별 집계

< 표 4 > 연 령 및 교육정도별 기혼부인

연 령	전 교육	교 육				
		불취학	국 민	중 학	고 등	대학이상
기혼부인						
총 수						
15 ~ 19						
20 ~ 24						
25 ~ 29						
30 ~ 34						
35 ~ 39						
40 ~ 44						
45 ~ 49						
50 ~ 54						
55 ~ 59						
60 ~ 64						
65+						

집계요령 : 1) 전국, 대도시, 시부, 군부  
 2) 가구표에서 집계  
 3) 연 1 회 집계 ( 央 年 )

< 표 5 >

연 도 별 초 혼 연 령

연 도	총 계	초 혼 연 령						평 균 초혼연령
		15세미만	15~19	20~24	.....	70세이상	미 상	
남 자								
⋮								
1982								
1981								
1980								
1979								
1978								
1977								
1976								
1975								
1974								
⋮								
1945								
1945 이 전								
여 자								
⋮								

집계요령 : 1) 전국, 시부, 군부  
 2) 가구료에서 집계  
 3) 연 1 회 집계 (연말)

< 표 6 >

월 별 출 생 수

성	총 계	1월	2월	3월	4월	5월	6월	7월	8월	9월	10월	11월	12월
계													
남													
여													

집계요령 : 1) 전국, 대도시, 시부, 군부  
 2) 동태조사표에서 집계  
 3) 분기별 집계

< 표 7 >

모의연령 및 출생아성별 출생자수

모 의 연 령	출 생 아 성 별		
	계	남	여
계			
15 ~			
15 ~ 19			
20 ~ 24			
25 ~ 29			
30 ~ 34			
35 ~ 39			
40 ~ 44			
45 ~ 49			
50+			

집계요령 : 1) 전국, 대도시, 기타시부, 군부  
 2) 동태조사표에서 집계  
 3) 분기별 집계

< 표 8 >

모의 연령 및 출산순위별 출생아수

출산당시 모의연령	총 계	출 산 순 위										이 상	
		1	2	3	4	5	6	7	8	9	10 이상		
여 자													
15세미만													
15 ~ 19													
20 ~ 24													
25 ~ 29													
30 ~ 34													
35 ~ 39													
40 ~ 44													
45 ~ 49													
50 +													
미 상													

집계요령 : 1) 전국, 대도시, 기타시부, 군부  
 2) 동태조사표에서 집계  
 3) 분기별 집계

< 표 9 >

모의 연령 및 교육정도별 출생아수

교육정도	계	0 무취학	1~6 국 민	7~9 중 학	10~12 고 등	13+ 대 학	미 상
연 령							
15 ~							
15 ~ 19							
20 ~ 24							
25 ~ 29							
30 ~ 34							
35 ~ 39							
40 ~ 44							
45 ~ 49							
50 +							
미 상							

집계요령 : 1) 전국, 대도시, 기타지방, 군부  
 2) 동태조사표에서 집계  
 3) 분기별 집계

< 표 10 >

월 별 사 망 자 수

	사망 총수	1월	2월	3월	4월	5월	6월	7월	8월	9월	10월	11월	12월
계													
남													
여													

집계요령 : 1) 전국, 시부, 군부  
 2) 동태조사표에서 집계  
 3) 분기별 집계



< 표 11 > 성, 연령 및 교육정도별 사망자

연령	계	불취학	국	중	고	대
남자						
0세						
1~4						
5~9						
10~14						
15~19						
20~24						
25~29						
30~34						
35~39						
40~44						
45~49						
50~54						
55~59						
60~64						
65~69						
70~74						
75~79						
80~84						
85+						
미상						
여자						

집계요령 : 1) 전국, 대도시, 기타시부, 군부  
 2) 동태조사표에서 집계  
 3) 분기별 집계

< 표 12 >

성·연령 및 직업별 사망자

성 및 직업	전 연령	연령				
		15세미만	15 ~ 19	.....	60 ~ 64	65 +
남 자						
대 분 류						
직업없음						
여 자						

- 집계요령 : 1) 전국, 시부, 군부  
 2) 동태조사표에서 집계  
 3) 년 1회 집계 (연말)  
 4) 경제활동의 직업분류중 대분류 참조

< 표 13 >

### 진 단 자 별 사 망

사 망 원 인	진 단 자		
	의 사	한 의 사	기 타
병 사  의 인 사  계  남  여			

- 집계요령 : 1) 전국, 시부, 군부  
 2) 별도요구가 있을 때까지 진단자별 사망만 집계  
 3) 분기별 집계

< 표 14 >

성·연령 및 배우관계별 사망자

연령	사망총수	미혼	유배우	사별	이혼	미상
남자						
계						
15세 미만						
15세 이상						
15 ~ 19						
}						
80 ~ 84						
85 +						
미상						
여자						

집계요령 : 1) 전국, 시부, 군부  
 2) 동태조사표에서 집계  
 3) 연 1회집계(연말)

< 표 15 >

### 성 및 연령별 사망원인

성 및 연령	사 망 원 인										
	계	병 사	외 인 사							기 타	
			1	2	3	4	5	6	7		8
계											
0											
1 ~ 4											
5 ~ 9											
10 ~ 14											
15 ~ 19											
20 ~ 24											
25 ~ 29											
30 ~ 34											
35 ~ 39											
40 ~ 44											
45 ~ 49											
50 ~ 54											
55 ~ 59											
60 ~ 64											
65 ~ 69											
70 ~ 74											
75 ~ 79											
80 ~ 84											
85 +											
남 자											
0											
1 ~ 4											
5 ~ 9											
10 ~ 14											
15 ~ 19											
20 ~ 24											
25 ~ 29											
30 ~ 34											
35 ~ 39											
40 ~ 44											
45 ~ 49											
50 ~ 54											
55 ~ 59											
60 ~ 64											
65 ~ 69											
70 ~ 74											
75 ~ 79											
80 ~ 84											
85 +											
여 자											
0											
1 ~ 4											
5 ~ 9											
10 ~ 14											
15 ~ 19											
20 ~ 24											
25 ~ 29											
30 ~ 34											
35 ~ 39											
40 ~ 44											
45 ~ 49											
50 ~ 54											
55 ~ 59											
60 ~ 64											
65 ~ 69											
70 ~ 74											
75 ~ 79											
80 ~ 84											
85 +											

집계요령 : 1) 전국, 시부, 군부  
 2) 동태조사표에서 집계  
 3) 연 1회 집계(연말)

< 표 16 >

### 일령 및 성별 유아사망

	당해년도 출생			전년도 출생		
	계	남	여	계	남	여
유아사망총수						
1일미만						
1~6일						
1						
2						
3						
⋮						
6						
7~27일						
7~13						
14~20						
21~27						
28~2달미만						
2~11달						
2달						
3달						
4달						
⋮						
11달						

- 집계요령 : 1) 전국, 시부, 군부  
 2) 동태조사표에서 집계  
 3) 연 1회집계(연말)

< 표 17 >

조산자별 유아사망

	조 산 자		
	병 원	조 산 원	기 타
유아사망총수			
1 일 미 만			
1 ~ 6 일			
7 ~ 27 일			
28 일 ~ 2 달 미 만			
2 ~ 11 달			
생 존 아			
출 생 아 총 수			

- 집계요령 : 1) 전국, 시부, 군부  
 2) 동태조사표에서 집계  
 3) 연 1회 집계 (연말)

< 표 18 >

월 별 혼 인

성 별	총수	1월	2	3	4	5	6	7	8	9	10	11	12
계													
남													
여													

- 집계요령 : 1) 전국, 대도시, 기타 시부, 군부  
 2) 동태조사표에서 집계  
 3) 분기별 집계



< 표 19 >

부처간의 혼인연령 분포

부연령	진연령	차 연 령									평 균 혼인연령
		15세 미만	15~ 19	20	21	22	...	29	30~ 34	35 이상	
전연령											
15세미만											
15~19											
20~24											
25											
26											
27											
⋮											
34											
35~39											
40+											
평 균 혼인연령											

- 집계요령 : 1) 전국, 대도시, 시부, 군부  
 2) 동태조사표에서 집계  
 3) 연 1 회 집계 (연말)

< 표 20 >

월 별 이 혼

성 별	이 혼 수	1월	2	3	4	5	6	7	8	9	10	11	12
계													
남													
여													

- 집계요령 : 1) 전국, 대도시, 기타 시부, 군부  
 2) 동태조사표에서 집계  
 3) 분기별 집계

< 표 21 >

### 혼인기간 및 연령별 이혼자

	총이혼수	1년미만	2년미만	3,4 ..... 9 ... 10-14,15-19	20년이상
전 연령 (부)					
15세 미만					
15 ~ 19					
20 ~ 24					
⋮					
65 ~ 69					
70 +					
미 상					
(처)					

- 집계요령 : 1) 전 국  
 2) 동태조사표에서 집계  
 3) 연 1회 집계 (연말)

< 표 22 >

### 생 존 자 녀 수 별 이 혼

부인연령	이 혼 총 수	생 존 자 녀 수								총 자녀수
		0	1	2	3	4	...	7이상	미 상	
전 연령										
15세 미만										
15 ~ 19										
⋮										
70 +										

- 집계요령 : 1) 전 국  
 2) 동태조사표에서 집계  
 3) 연 1회 집계(연말)

< 표 23 >

월 별 , 성 별 이 동 인 구

	계	1월	2	3	4	5	6	7	8	9	10	11	12
전 입 자													
계													
남													
여													
전 출 자													
계													
남													
여													

- 집계요령 : 1) 전국, 대도시, 기타시부, 군부  
 전입자는 현거주지=목적지  
 ( 전출자는 전출전 거주지=출발지 기준 )  
 2) 변동보고서에서 집계  
 3) 분기별 집계

< 표 24 >

출발지및현거주지별인구이동

출 발 지	목 적 지(현 거 주 지)			
	계	서울,부산,대구,인천	기 타 시	군 부
도 간 이 동				
계				
서울,부산,대구,인천				
기 타 시				
군 부				
미 상				
도 내 이 동				
계				
서울,부산,대구,인천				
기 타 시				
군 부				
미 상				

- 집계요령 : 1) 전입자만으로 집계  
 2) 변동보고서에서 집계  
 3) 분기별 집계

< 표 25 > 연령, 성 및 이동종류별 이동인구 및 비이동인구

연령및성 (현거주지)	계	비 이 동 인 구	조 사 구 내 이 동	같은구 (시군) 내이동	도 내 이 동				도 간 이 동				외국	
					계	대 도시	시부	군부	계	대 도시	시부	군부		
전 국														
계														
0 ~ 5														
6 ~ 12														
13 ~ 15														
16 ~ 18														
19														
20 ~ 24														
25 ~ 29														
30 ~ 34														
35 ~ 39														
40 ~ 44														
45 ~ 64														
65 ~ 74														
75 +														
남 자														
여 자														
대 도시														
남 자														
여 자														
남 자														
여 자														
남 자														
여 자														

집계요령 : 1) 전입자로 집계  
 2) 변동보고서에서 집계  
 3) 분기별 집계

< 표 26 >

이동형태 및 이동사유별 이동인구

이동사유	가 구 수	가 구 원 수	개 인 이 동
총 이 동 수 직 장 관 계 취 업 후 은 전 직 구 직 전 근 새 로 운 사 업 시 작 직 장 이 의 교 통 관 계 입 대 및 제 대 주 택 관 계 부 득 이 한 이 사 가 족 관 계 혼 인 이 혼 등 가 구 원 이 동 교 육 자 신 의 교 육 가 족 의 교 육 건 강 기 타 무 응 답 미 상			

- 집계요령 : 1) 전국, 대도시, 기타시부, 군부  
 2) 전입자만으로 집계  
 3) 변동보고서에서 집계  
 4) 분기별 집계



## 8. 조 사 표

※ 통계법 제9조에 의하여 통계목적 이외의 사용을 금함

--	--	--

조사구번호	가구번호	가구주이름
- - -	- - -	
조사담당자	확인 및 가구표성리	담
	내검요위	닝

인구동태표본조사표  
(198년 □ 월분)

① 본적	시도	구시군	리	번호	호수성명:
------	----	-----	---	----	-------

② 전수	③ 모의이름	④ 의가구번호	⑤ 출생아름	⑥ 출생년월일	⑦ 성별	⑧ 출생의령	⑨ 출산순위	⑩ 임신기간	⑪ 모의교육정도	⑫ 조		
										1	2	3
1				198년 월 일	1남 2여			개월		1. 병원 2. 조산원 3. 기타( )		
2				198년 월 일	1남 2여			개월		1. 병원 2. 조산원 3. 기타( )		

① 본적	시도	구시군	리가	번호	호수성명:
------	----	-----	----	----	-------

② 전수	③ 사망자름	④ 가구원호	⑤ 성별	⑥ 생년월일	⑦ 사망년월일	⑧ 사망자의업종	⑨ 직업	⑩ 혼인상태	⑪ 진단자			⑫ 사망원인
									1. 의사 2. 한의사 3. 기타( )	1. 의사 2. 한의사 3. 기타( )	1. 병원 2. 조산원 3. 기타( )	
1			1남 2여	198년 월 일	198년 월 일							
2			1남 2여	198년 월 일	198년 월 일							

③ 혼인

② 전수	③ 이혼			④ 이혼			⑤ 이혼			⑥ 이혼		
	③ 이혼	④ 이혼	⑤ 이혼	⑥ 이혼	⑦ 이혼	⑧ 이혼	⑨ 이혼	⑩ 이혼	⑪ 이혼	⑫ 이혼	⑬ 이혼	⑭ 이혼
1												
2												

④ 이혼

② 전수	③ 이혼	④ 이혼	⑤ 이혼	⑥ 이혼	⑦ 이혼	⑧ 이혼	⑨ 이혼
1							
2							

※ 통계법 제9조에 의하여 통계목적 이외의 사용을 금함.

가 구 표		①	②	③	④	⑥	⑦	⑧	매										
198 . . . . . 작성		조사구 번호	구역 번호	거처 번호	가구 번호	거처 구분	1. 독립주택 2. 연립주택 3. 아파트 4. 비거주용건물 내의 주택 5. 주택이외의 거처	소유 관계	1. 자가 2. 전세 3. 보증부월세 4. 식월세 5. 월세 6. 기타	8. 구역 구분	1. 도시가계 2. 경제활동 3. 인구동태	중 매							
		⑤							⑨		1. 봉급자 2. 노무자	3. 근로자외 4. 부적격							
⑩	⑪	⑫	⑬	⑭			⑮	⑯		⑰		⑱	⑳	㉑	㉒				
번호	이 름	가구주 관계	성 별	생 년 월 일			학 력	배 우 관계	초 혼		출 산 계 획	임 신 여 부		전 입		비 고	가 계 조 사 부 적 격 사 유		
				년	월	일			년	월		남	여	년	월			결 과	년
01				1	년	월	일	1. 유배우 2. 사별 3. 이혼 4. 미혼			19	년	월	1. 출생 2. 중절 3. 사산 4. 기타	19	년	월	일	
02				1	년	월	일	1. 유배우 2. 사별 3. 이혼 4. 미혼			19	년	월	1. 출생 2. 중절 3. 사산 4. 기타	19	년	월	일	
03				1	년	월	일	1. 유배우 2. 사별 3. 이혼 4. 미혼			19	년	월	1. 출생 2. 중절 3. 사산 4. 기타	19	년	월	일	㉓ 비고(거처 및 가구특징 등)
04				1	년	월	일	1. 유배우 2. 사별 3. 이혼 4. 미혼			19	년	월	1. 출생 2. 중절 3. 사산 4. 기타	19	년	월	일	
05				1	년	월	일	1. 유배우 2. 사별 3. 이혼 4. 미혼			19	년	월	1. 출생 2. 중절 3. 사산 4. 기타	19	년	월	일	
06				1	년	월	일	1. 유배우 2. 사별 3. 이혼 4. 미혼			19	년	월	1. 출생 2. 중절 3. 사산 4. 기타	19	년	월	일	
07				1	년	월	일	1. 유배우 2. 사별 3. 이혼 4. 미혼			19	년	월	1. 출생 2. 중절 3. 사산 4. 기타	19	년	월	일	

경제기획원조사통계국

※ 통계법 제9조에 의하여 통계목적 이외의 사용을 금함.

변동보고서 198    년    월분	① 조사구 번호 [ ][ ][ ][ ][ ][ ]	② 가구표성리 조사담당    지도구책임자    담당	③ 현황 가구    전 원    증 가    감 소    금 월 인구	④ 출생수    사망수
-------------------------	--------------------------------	---------------------------------	--	--------------

1 가구 증가 (⑤ 가구 ⑥ 명)							2 가구 감소 (⑬ 가구 ⑭ 명)														
⑦	⑧	⑨	⑩	⑪		⑬	⑮	⑯	⑰	⑱		⑲									
가 구 번 호	가 구 주 이 름	가 구 원 수	변 동 일 자	전	거 주 지	이 동 사 유	가 구 번 호	가 구 주 이 름	가 구 원 수	변 동 일 자	복 적 지	이 동 사 유	구 분								
1	---		198	시	구	1.	1	---		198	시	구	1.								
2	---		198	시	구	2.	2	---		198	시	구	2.								
3	---		198	시	구	1.	3	---		198	시	구	1.								
4	---		198	시	구	2.	4	---		198	시	구	2.								
5	---		198	시	구	1.	5	---		198	시	구	1.								
6	---		198	시	구	2.	6	---		198	시	구	2.								
7	---		198	시	구	1.	7	---		198	시	구	1.								
조사구내의 가구로서 분 가 한 가 구 수		가 구		※ 1. ⑬ 구분란은 전입, 전입누락, 착오등을 기입 2. ⑤란에는 조사구내 분가한 가구수도 합계됨							조사구내의 가구로서 통 합 된 가 구 수		가 구		※ 1. ⑲ 구분란에는 전출, 전출누락, 사망, 착오등을 기입 2. ⑬란에는 조사구내 통합된 가구수도 합계됨						

경제기획원 조사통계국

3 가구원 증가(23 명)

구분	24 가구번호	25 가구원번호	26 이름	27 가구주와의 관계	28 성별	29 생년월일			30 학력	31 보육 관련 과목 취득 유무	32 혼인		33 출산 계획		34 임신 여부		35 전거 주지	36 전입 일자	37 이동 사유	38 구분		
						년	월	일			년	월	남	여	년	월					결	과
1	- - -				1남 2여	19	년	월	일	1 2 3 4 유 무 취 득 유 무	19	년	월	19	년	월	1 2 3 4 출 생 사 기 생 결 산 타	시 도	구 시 군	19	1. 2.	
2	- - -				1남 2여	19	년	월	일	1 2 3 4 유 무 취 득 유 무	19	년	월	19	년	월	1 2 3 4 출 생 사 기 생 결 산 타	시 도	구 시 군	19	1. 2.	
3	- - -				1남 2여	19	년	월	일	1 2 3 4 유 무 취 득 유 무	19	년	월	19	년	월	1 2 3 4 출 생 사 기 생 결 산 타	시 도	구 시 군	19	1. 2.	
4	- - -				1남 2여	19	년	월	일	1 2 3 4 유 무 취 득 유 무	19	년	월	19	년	월	1 2 3 4 출 생 사 기 생 결 산 타	시 도	구 시 군	19	1. 2.	
5	- - -				1남 2여	19	년	월	일	1 2 3 4 유 무 취 득 유 무	19	년	월	19	년	월	1 2 3 4 출 생 사 기 생 결 산 타	시 도	구 시 군	19	1. 2.	
6	- - -				1남 2여	19	년	월	일	1 2 3 4 유 무 취 득 유 무	19	년	월	19	년	월	1 2 3 4 출 생 사 기 생 결 산 타	시 도	구 시 군	19	1. 2.	
7	- - -				1남 2여	19	년	월	일	1 2 3 4 유 무 취 득 유 무	19	년	월	19	년	월	1 2 3 4 출 생 사 기 생 결 산 타	시 도	구 시 군	19	1. 2.	
8	- - -				1남 2여	19	년	월	일	1 2 3 4 유 무 취 득 유 무	19	년	월	19	년	월	1 2 3 4 출 생 사 기 생 결 산 타	시 도	구 시 군	19	1. 2.	
9	- - -				1남 2여	19	년	월	일	1 2 3 4 유 무 취 득 유 무	19	년	월	19	년	월	1 2 3 4 출 생 사 기 생 결 산 타	시 도	구 시 군	19	1. 2.	
10	- - -				1남 2여	19	년	월	일	1 2 3 4 유 무 취 득 유 무	19	년	월	19	년	월	1 2 3 4 출 생 사 기 생 결 산 타	시 도	구 시 군	19	1. 2.	

\* 38 구분란은 출생, 전입, 누락등으로기입

4 가 구 원 감 소 (㉞ 명)						
가 구 번 호	가 구 원 번 호	이 름	변 동 일 자	목 적 지	이 동 사 유	구 분
2			198	시 도 구 시 군	1. 2.	
3			198	시 도 구 시 군	1. 2.	
4			198	시 도 구 시 군	1. 2.	
5			198	시 도 구 시 군	1. 2.	
6			198	시 도 구 시 군	1. 2.	
7			198	시 도 구 시 군	1. 2.	
8			198	시 도 구 시 군	1. 2.	
9			198	시 도 구 시 군	1. 2.	
10			198	시 도 구 시 군	1. 2.	

5 가 구 표 수 정 요 구 사 항				
가 구 번 호	가 구 원 번 호	가 구 표 란 번 호	수 정 할 내 용	
			수 정 전 내 용	수 정 후 내 용
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				

\* ㉞ 구분란은 전출, 전출수락, 사망, 착오등을 기입

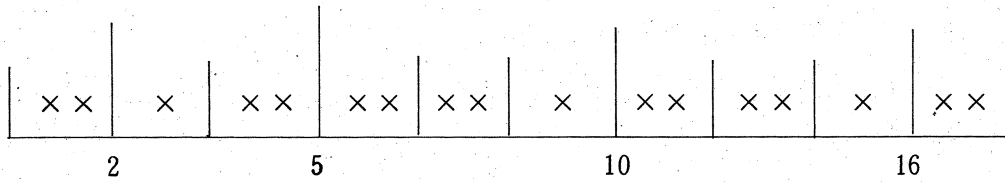


## 9. TAPE 표 찰 방 법



- 작업을 완료한 후에는 BACK-UP TAPE FILE을 보관하여야 하므로 전산실에 작업이 끝났음을 알리고 TAPE FILE 상태를 DITTO 하여 확인하고 보관할 것.
- BACK-UP TAPE 작성시 LABEL은 다음과 같은 CODE형식에 따라 작성한다.
- 각과 고유번호
  - 자료처리과 : A
  - 자료관리과 : B
  - 통계분석과 : C
  - 유통통계과 : D
  - 산업통계과 : E
  - 인구통계과 : F
  - 사회통계과 : G
  - 조사관리과 : H
  - 기준과 : L
  - 서무과 : N

## TAPE 작업 전산화 CODE



年度	課	業務名	過程	차수	RUN TEST	USER CODE	도·월 分期	USER 使用	REEL 수
				0.1					2-1
				}			↓		2-1
				99			월 01~12		
					R-RUN				
					T-TEST				

- 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 : BACKUP
- MS : MASTER
- CV : CONVERT