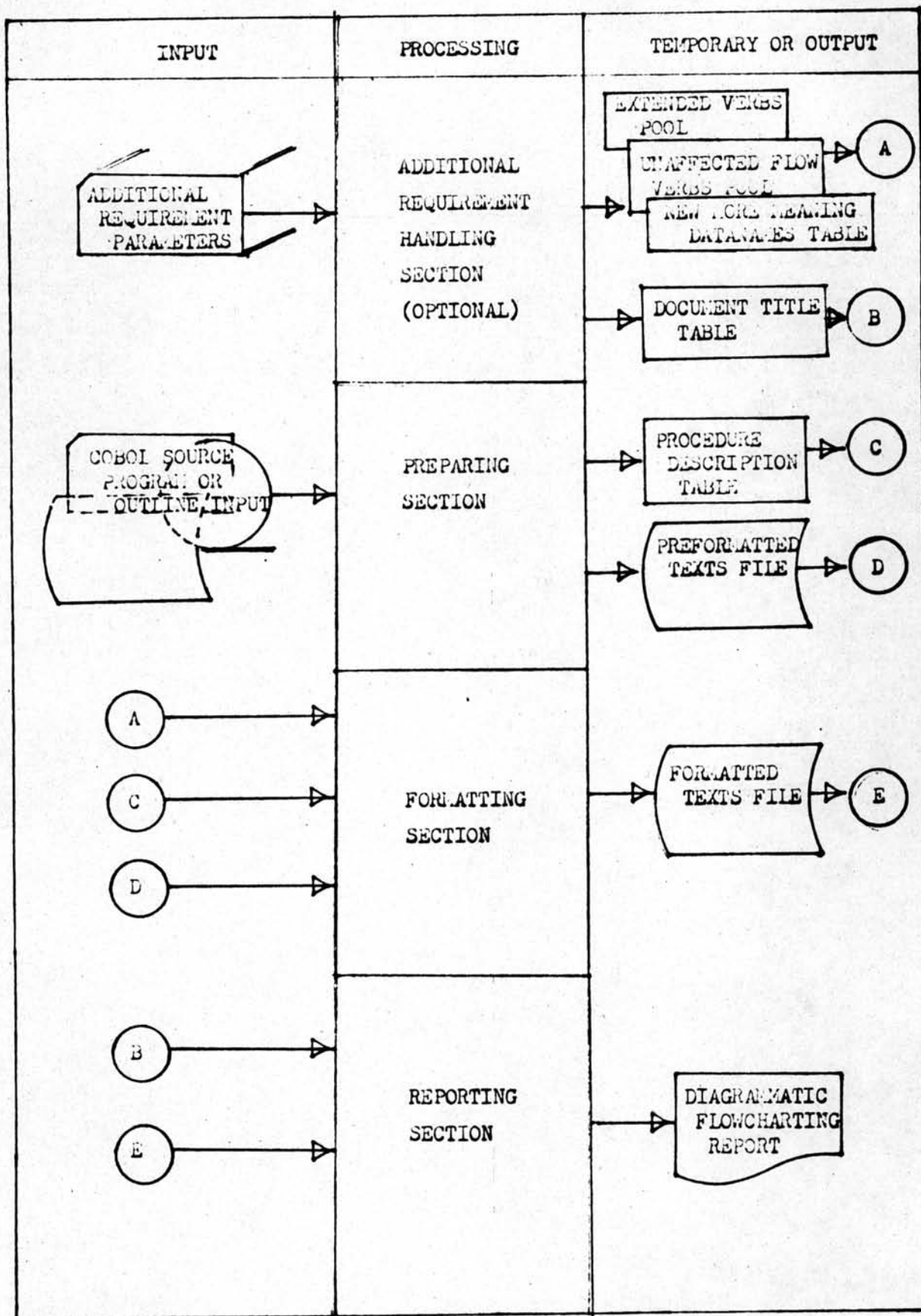


REFERENCE

- Donald E. Knuth Fundamental of Algorithms Addison-Wesley publishing
Company, 1973
- Edsger W. Dijkstra & C.A.R. Hoare Structured Programming Academic
Press, 1972
- Edsger W. Dijkstra A Discipline of programming Prentice-Hall, 1976
- S.R. Mixon Handbook of Data Processing Administration, Operation and
Procedures A Division of American Management Association,
Newyorks, 1976
- I. C. T. Data Processing Standards and Procedures, Volume IV
- I. B. M. American National Standard COBOL Seven Edition, 1976
- Burroughs COBOL ANSI-74, Reference Manual

APPENDIX A

TOTAL VIEWS OF THE PROGRAM



APPENDIX B
OUTLINE FLOWCHART OF THE PROGRAM

PROGRAM LOGIC FLOWCHART OF THE TOOL

I THINK I HAVE NO ANY METHODS TO EXPLAIN THIS TOOL
BETTER THAN THE OUTLINE LOGIC FLOWCHART. NOW LET IT EXPLAINS
ITSELF.

.....
MAINLINE

SECTION # 001
.....

THIS IS THE MAIN PROCEDURE OF THE FLOWCHART GENERATOR
ITS DIVIDED INTO 4 MAIN SECTION.

THE FIRST SECTION : OPTIONAL ADDITION FEATURE
1) VERBS ADDED FEATURE
2) VERBS DELETED FEATURE
3) DATANAME REPLACED FEATURE
4) DOCUMENTATION TITLE FEATURE

THE SECOND SECTION: - EDITED AND SELECTED SOURCE FILE
- PROCEDURE NAME TABLE CREATED

THE THIRD SECTION : - PICK UP COBOL OR OUTLINE SOURCE RECORD
- CONCATENATE IT TO STRING
- CUT THE STRING TO STATEMENTS
- VERB PATTERN PROCESS AND FORMATTED
- FORMATTED STATEMENT FILE CREATED

THE LAST SECTION : - PROGRAM LOGIC FLOWCHART PRINTED

(FLOW BEGIN)

I
V
(NEXT PAGE)

FORMER CHAIRMAN OF THE NATIONAL ASSOCIATION OF PROFESSIONAL ENGINEERS

IF
< YOU WANT SOME ADDITIONAL FEATURE DESCRIBED IN >
< THE FIRST SECTION >

NO

YES

UNTIL COMPLETE YOUR REQUIREMENT
PERFORM THE-FIRST-PROCEDURE (S002)

UNTIL NEW SOURCE FILE ALREADY CREATED
PERFORM THE-SECOND-PROCEDURE (S003)

UNTIL FORMATTED STATEMENT FILE HAS BEEN
CREATED
PERFORM THE-THIRD-PROCEDURE (S004)

UNTIL THE PROGRAM LOGIC FLOWCHART HAS BEEN
PRINTED COMPLETELY
PERFORM THE-LAST-PROCEDURE (S005)

(FLOW END)

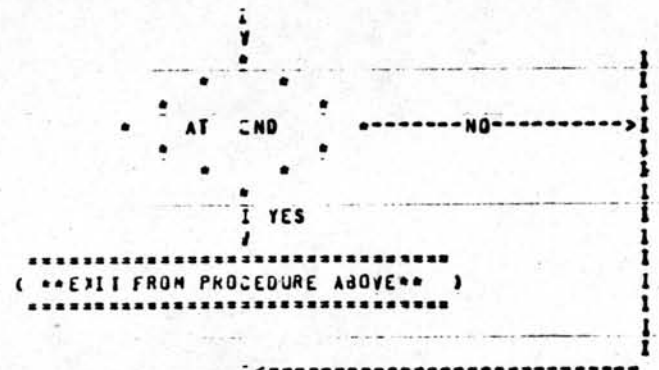
*****END OF MAINLINE PART*****

.....
THE-FIRST-PROCEDURE SECTION # 002
.....

REMARKS :- WHEN YOU WANT THESE ADDITIONAL FEATURE YOU MUST
ENTER "YES" UPON CONSOLE.

- WHAT FEATURE YOU WANT?
- TYPE 1 : VERBS ADDED
- TYPE 2 : VERBS DELETED
- TYPE 3 : DATANAME REPLACED
- TYPE 4 : DOCUMENT TITLE

/ READ YOUR REQUIREMENT INPUT PARAMETER DATA /



< DEPENDING YOUR-REQUIREMENT >

- I
- I-----01----->(P001)
- I
- I-----02----->(P003)
- I
- I
- I-----03----->(P005)
- I
- I
- I-----04----->(P007)
- I

(NEXT PAGE)

```

      ELSE
      /
.....
( **EXIT FROM PROCEDURE ABOVE** )
.....

```

 VERB-ADDED-PROCEDURE PROCEDURE # 001

```

INPUT PARAMETER FORMAT (START AT COLUMN 9)
I
I-----> ADD-VERB LIST
I-----> <VERB LIST>
I-----> I---> VERB1 VERB2 VERB3 ....
I         I---> "   "   "   "   ....
I         I---> "   "   "   "   ....
I         .
I
I-----> ADD-VERB END
I
V

```

```

I-----I
II      UNTIL "ADD-VERB END" FOUND          II
IV                                             VI
-----
II PERFORM VERB-ADDED-POOL-CREATED ( P002 ) II
-----

```

```

      /
.....
( S002 )
.....

```

 VERB-ADDED-POOL-CREATED PROCEDURE # 002

(NEXT PAGE)


```

-----
/ READ <VERB LIST> FROM YOUR SUBMITTED INPUT /
/ DATA /
-----

```

```

I
/
I MOVE VERB NAME TO VERB ADDED POOL ACCORDING I
I TO ITS POSITION I
/
I
/

```

```

-----
VERB-DELETED-PROCEDURE          PROCEDURE # 003
-----

```

INPUT PARAMETER FORMAT (START AT COLUMN 0)

```

I
I-----> DELETE-VERB LIST
I-----> <VERB LIST>
I-----> I---> VERB1 VERB2 VERB3 ....
I-----> I---> " " " " ....
I-----> V---> " " " " ....
I
I
I-----> DELETE-VERB END
V

```

```

-----
I-----I
II UNTIL "DELETE-VERB END" FOUND II
IV VI
-----
II PERFORM VERB-DELETED-POOL-CREATED ( P004 ) II
-----

```

```

I
V
*****
( S002 )
*****

```

```

-----
VERB-DELETED-POOL-CREATED      PROCEDURE # 004
-----

```

(NEXT PAGE)

```

-----
/ READ <VERB LIST> FROM YOUR SUBMITTED INPUT /
/ DATA /
-----

```

```

-----
I MOVE VERB NAME TO VERB DELETED POOL ACCORDING I
I TO ITS POSITION I
-----

```

```

-----
DATANAME-REPLACED-PROCEDURE PROCEDURE # 005
-----

```

```

INPUT PARAMETER FORMAT (START AT COLUMN 9)

```

```

I
I-----> MATCH-NAME LIST
I-----> <REPLACED DATANAME> <REPLACING DATANAME>
I
I [----> "
I [----> "
I
I v
I
I
I-----> MATCH-NAME END
I
I
I

```

```

-----
I VARYING TABLE POINTER FROM 1 INCREMENT BY 1 II
I UNTIL "MATCH-NAME END" FOUND II
IV YI
-----
II PERFORM NAME-REPLACED-TABLE-CREATED ( P006 ) II
-----

```

```

:
/
-----
( S302 )
-----

```

```

-----
NAME-REPLACED-TABLE-CREATED PROCEDURE # 005
-----

```

```

(NEXT PAGE)

```

```

-----
/ READ PAIR OF REPLACED AND REPLACING DATANAME /
/ FROM SUBMITTED INPUT DATA /
-----

```

I
V

```

-----
I MOVE REPLACED AND REPLACING DATANAME TO TABLE I
I ACCORDING TO ITS POINTER I
-----

```

I
V

```

-----
DOCUMENTATION-TITLE-PROCEDURE PROCEDURE # 007
-----

```

INPUT PARAMETER FORMAT

```

I
I-----> COMMENT-LIST (COL. 8)
I-----> <COMMENT TITLE TEXT>
I-----> I-----> <LINE CONTROL> (COL. 1-5)
I-----> I I--> EJECT (FOR SKIP PAGE)
I-----> I I--> SKIP1 (FOR 1 LINE)
I-----> I I--> SKIP2 (FOR 2 LINES)
I-----> I I--> SKIP3 (FOR 3 LINES)
I-----> I I--> <BLANK> (FOR 1 LINE DEFAULT)
I-----> I I--> FOOT (FOR FOOT OF PAGE)
I
I
I
I
I
I-----> <RESERVE>
I-----> HYPHEN (COL. 7 ; FOR CONTINUATION COMMENT)
I-----> <TEXT> (CJL 8-72)
I-----> <RESERVE> (COL 73-80)
I
I
I
I-----> COMMENT END
V

```

```

-----
II VARYING TABLE POINTER FROM 1 INCREMENT BY 1 II
II UNTIL "DOCUMENT END" FOUND II
IV VI
-----
II PERFORM DOCUMENT-TABLE-CREATED ( P008 ) II
-----

```

I
V
(NEXT PAGE)

THE SECOND PROCEDURE SECTION # 003

- PROCEDURE PART OF SOURCE FILE SELECTED FOR LOGIC FLOW GENERATING (NEW EDITED SOURCE FILE CREATED)
- PROCEDURE DESCRIPTION TABLE (NAME, TYPE, NUMBER) CREATED
- 4Y COMMENT (X AT COL 7 IN SOURCE FILE) INCLUDED
- COBOL COMMENT (* AT COL 7) RECLUDED

/ READ SOURCE FILE UNTIL "PROCEDURE DIVISION" /
/ FOUND /

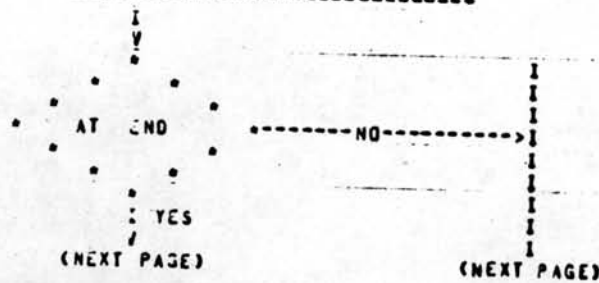
I
/

I	UNTIL END-OF SOURCE FILE	I
II		II
IV		VI
II	PERFORM PROCTAB-AND-NEWSOURCE-CREATED (P009	II
II)	II

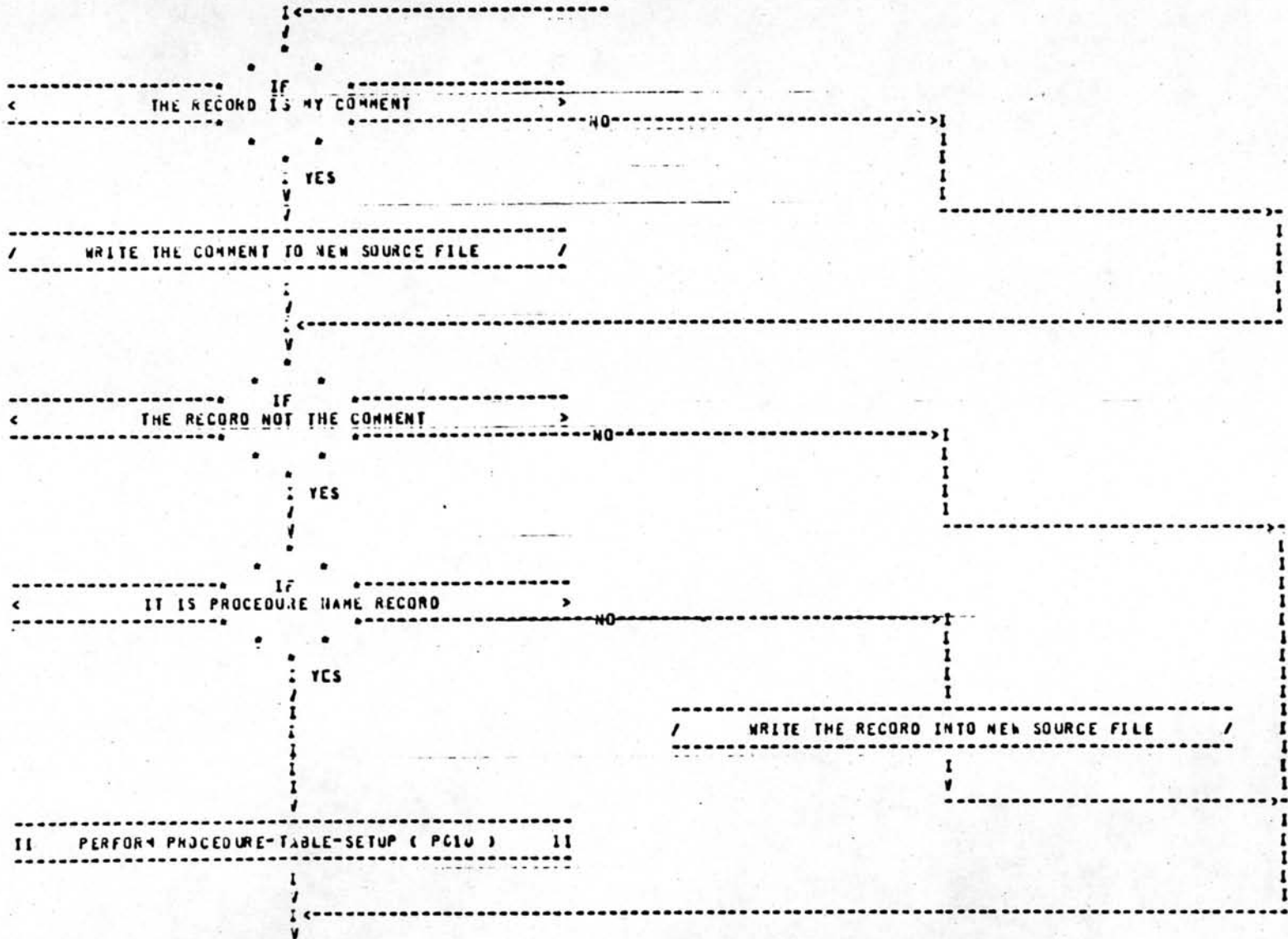
(**EXIT FROM PROCEDURE ABOVE**)

PROCTAB-AND-NEWSOURCE-CREATED PROCEDURE # 009

/ READ SOURCE FILE /



(**EXIT FROM PROCEDURE ABOVE**)



(NEXT PAGE)

PROCEDURE-TABLE-SETUP

PROCEDURE # 010

2 TYPES OF PROCEDURE NAME : PARAGRAPH (P) AND SECTION (S)

```

I  MOVE PROCEDURE NAME TYPE AND NUMBER INTO  I
I  PROCEDURE DESCRIPTION TABLE            I
  
```



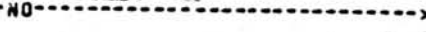
```

/  WRITE THE PROCEDURE RECORD INTO NEW SOURCE  /
/  FILE                                         /
  
```



```

          .   IF   .
<  HAVE REMAIN TEXT FOLLOW UP THE PROCEDURE NAME  >
  
```



. YES



```

/  WRITE THE REMAIN TEXT INTO NEW SOURCE FILE  /
  
```



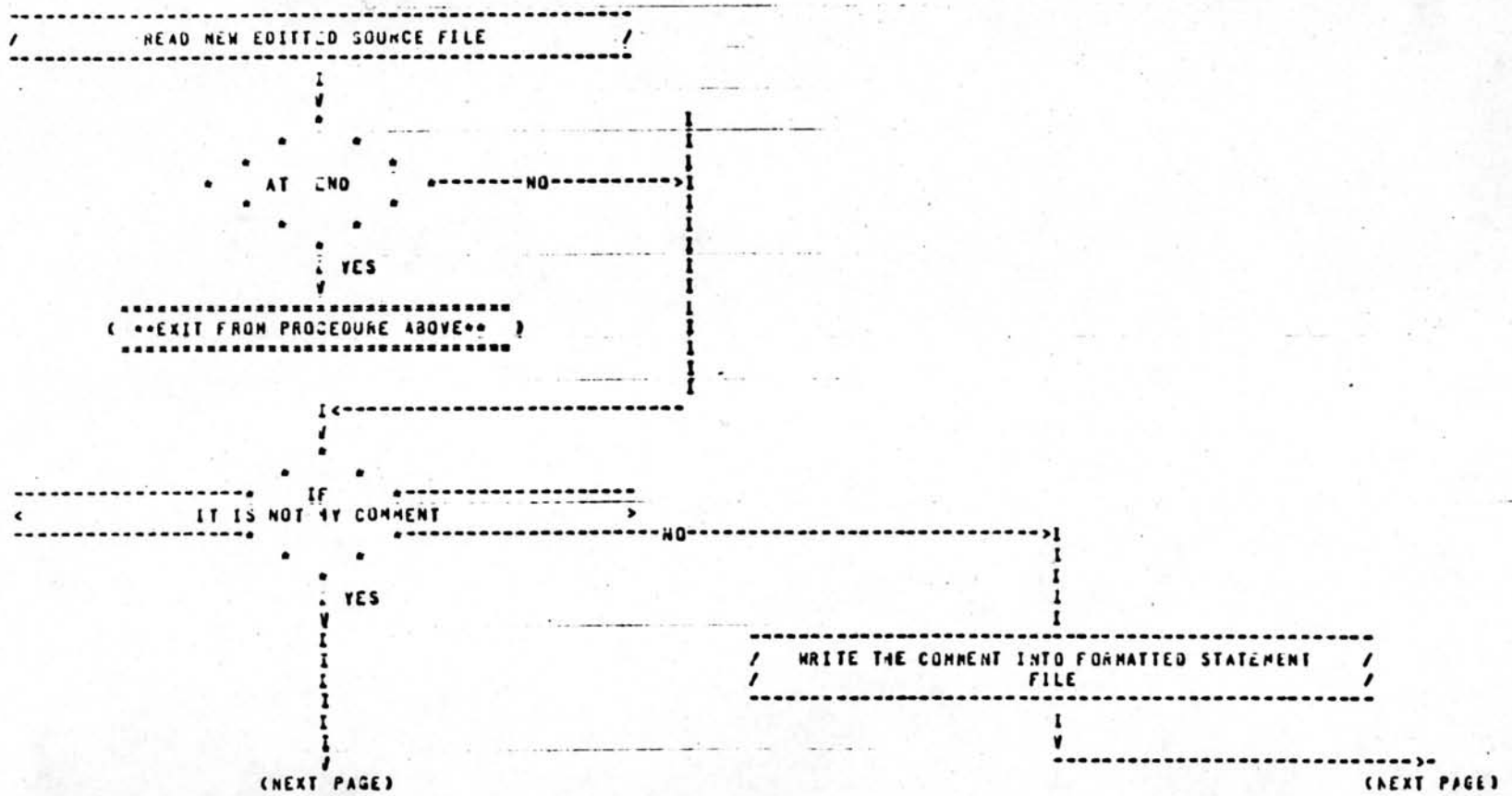
*****END OF THE SECOND--PROCEDURE*****

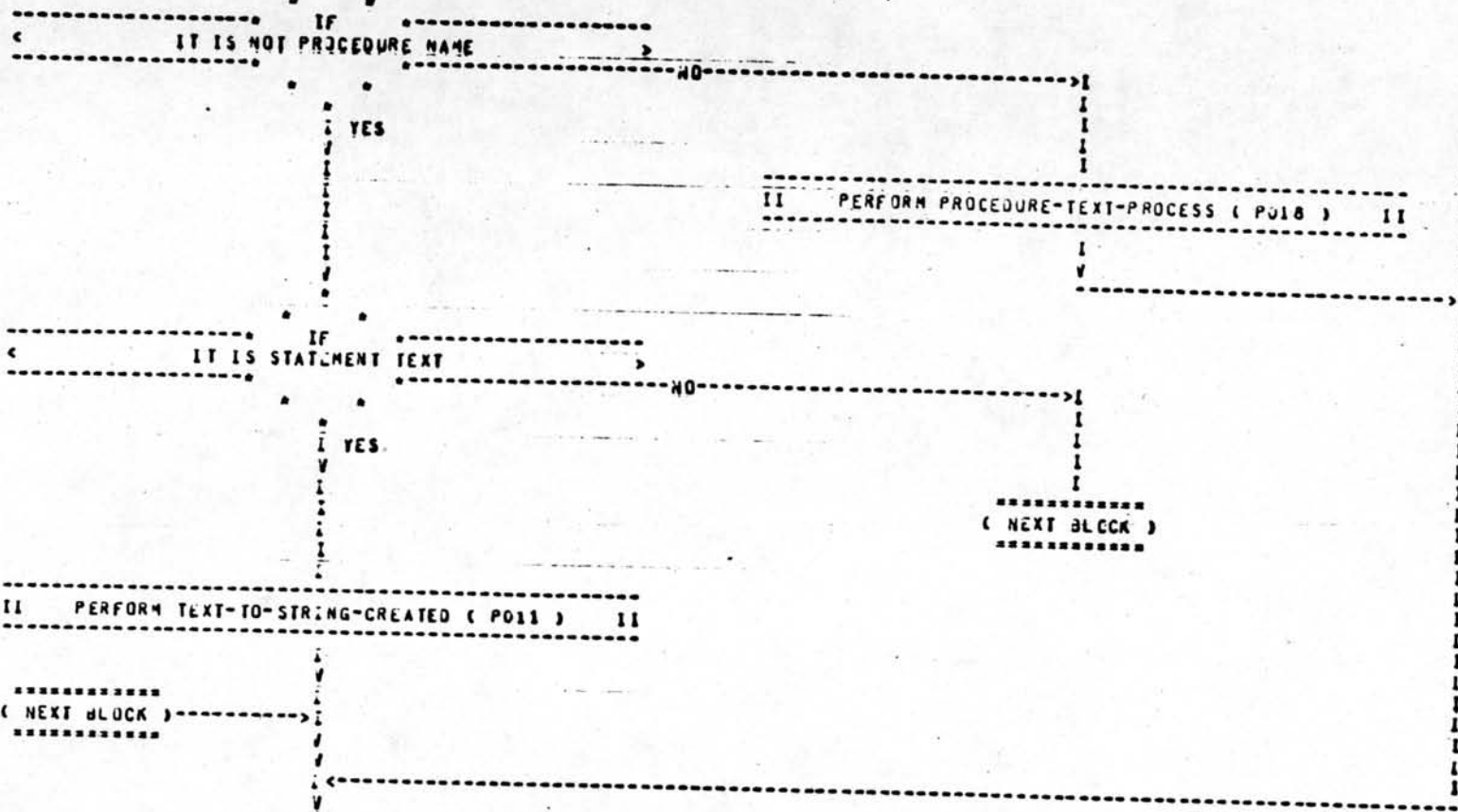
THE-THIRD-PROCEDURE

SECTION # 004

(NEXT PAGE)

- CONCATENATE TEXT INTO A STRING
- UNSTRING IT INTO MANY STATEMENTS
- VERBS TYPE PROCESS
- DATANAME REPLACED
- FORMATTED STATEMENT FILE CREATED

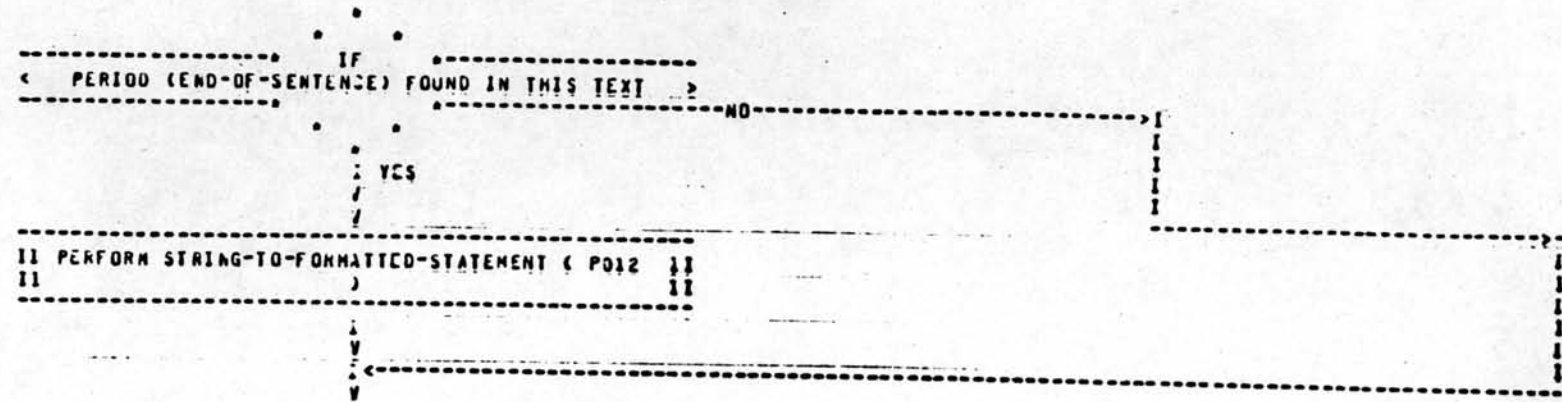




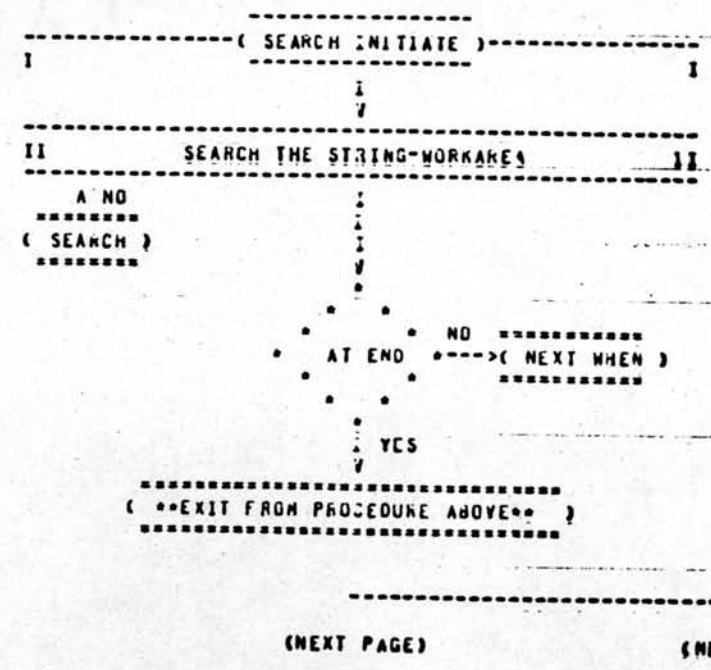
TEXT-TO-STRING-CREATED PROCEDURE # 011

I MOVE TEXT FROM EDITED SOURCE RECORD INTO I
I STRING-WORKAREA I

(NEXT PAGE)



STRING-TO-FORMATTED-STATEMENT PROCEDURE # 012



(NEXT PAGE)

(NEXT PAGE)

II PERFORM PROCEDURE-NUMBER-SEARCH (P017) II

A : YES
I NO
(NEXT WHEN)

THE WORD IS PROCEDURE NAME

WHEN

II PERFORM DATANAME-REPLACED-OR-NOT (P016) II

A : YES
I NO
(NEXT WHEN)

THE WORD IN STRING-HOKKAREVA IS DATANAME

WHEN

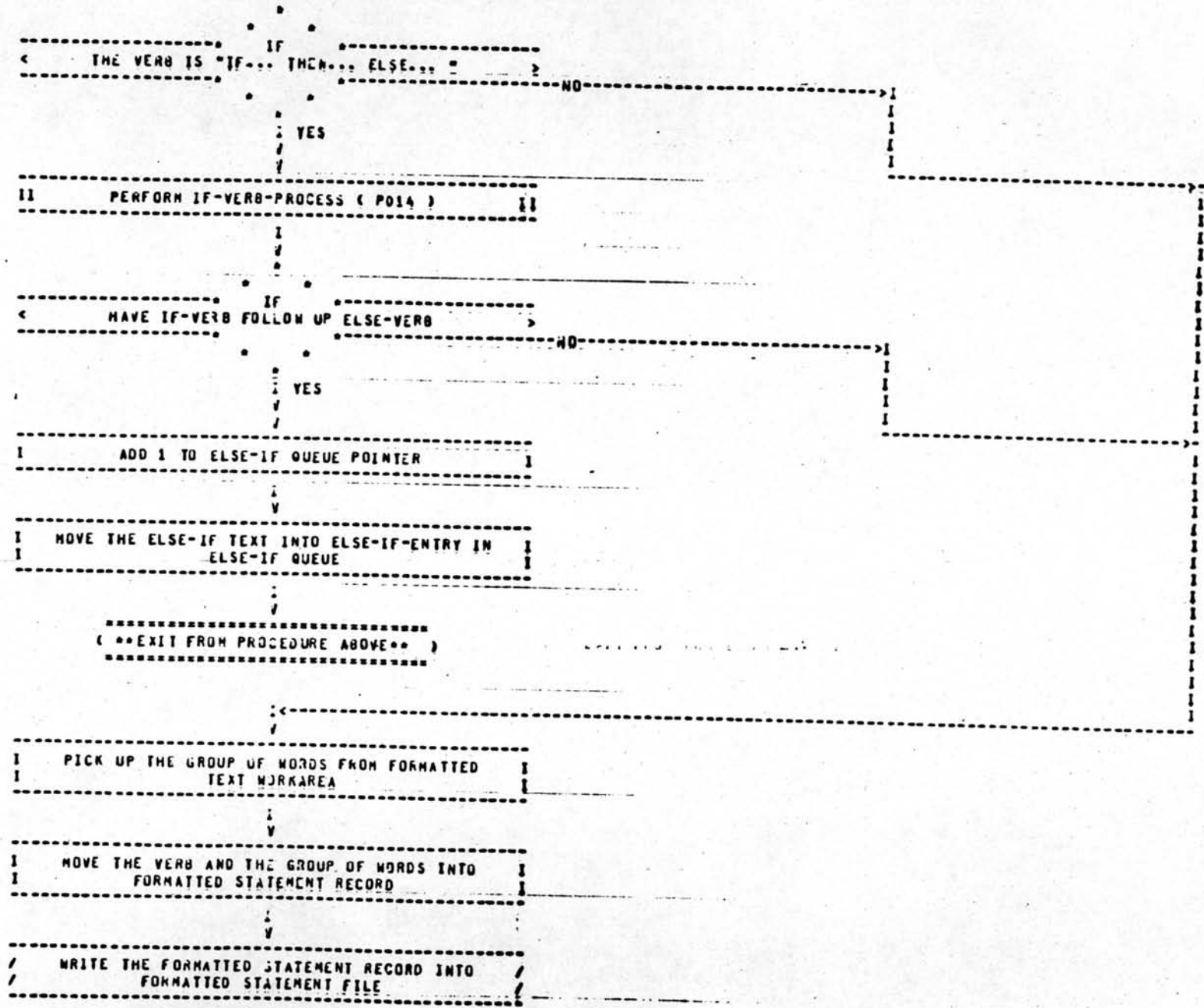
II PERFORM VERB-TYPE-PROCESS (P013) II

A : YES
I NO
(NEXT WHEN)

THE WORD IN STRING-HOKKAREVA IS VERB

WHEN





(NEXT PAGE)

IF-VERB-PROCESS

PROCEDURE # 014

USING THE STACK TECHNIQUES TO MAHAGE NESTED IF. THE NESTED STATEMENTS IS MUCH-LIKE SPECIFYING SUBORDINATE ARITHMETIC EXPRESSION ENCLOSED IN PARENTHESIS. "IF" VERB IS LIKE LEFT PARENTHESIS AND "ELSE" VERB LIKE RIGHT PARENTHESIS.

I INITIAL STACK LEVEL POINTER BY ZEROES I

V

I (SEARCH INITIATE) I

V

II SEARCH THE STRING-MOKKAEA II

A NO

 (SEARCH)

V

V

V

• • • NO *****
 • AT END •---->(NEXT WHEN)
 • • • *****

I YES

V

I-----I

II VARYING LEVEL POINTER FROM 1 INCREMENT BY 1 II
 II UNTIL NO MORE TEXT IN STACK II

IV VI

II PERFORM STACK-DECOMPOSE (P015) II

V

----->I

(NEXT PAGE)

(NEXT PAGE)

WHEN
 < "IF" VERB FOUND AGAIN INCREMENT LEVEL POINTER >
 BY 1

I NO
 V
 (NEXT WHEN)
 YES

I GIVE TEXT AFTER "IF" VERB TO "IF" CONDITION
 I TEST STACK-ENTRY (LEVEL POINTER) I

WHEN
 < "THEN" VERB OR ANY IMPERATIVE VERB FOUND >

I NO
 V
 (NEXT WHEN)
 YES

I GIVE THE STATEMENTS TO "IF" CONDITION TRUE
 I STACK-ENTRY (LEVEL POINTER) I

(SEARCH)
 A NO
 WHEN
 < "ELSE" VERB FOUND >

I YES

I GIVE THE STATEMENT AFTER "ELSE" VERB TO "IF"
 I CONDITION FALSE STACK-ENTRY (LEVEL POINTER) I

(NEXT PAGE)

(NEXT PAGE)

```

-----
| DECREMENT LEVEL POINTER BY 1 |
-----
|
|
|-----
|
|
|

```

```

-----
STACK-DECOMPOSE

```

```

PROCEDURE # 015
-----

```

AFTER IF, THEN, ELSE STACK ARE ALREADY CREATED, NOW WE MUST DECOMPOSE THE TEXTS IN EACH STACK WITH QUEUEING PROCESS. NOTE THAT :- THESE STACK IN PHYSICAL IS THE AREA IN THE STRING-WORKAREA, WE ONLY USED THE POINTERS, BEGIN AND END POINTERS (SUBSCRIPT WITH EACH LEVEL) REPRESENT EACH ENTRY IN EACH STACKS.

```

-----
| REPROCESS VERB-TYPE-PROCESS PROCEDURE FROM |
| "IF" CONDITION STACK-ENTRY (LEVEL POINTER) |
-----

```

```

|
|
|

```

```

-----
| REPROCESS VERB-TYPE-PROCESS PROCEDURE FROM |
| "IF" CONDITION TRUE STACK-ENTRY (LEVEL |
| POINTER) |
-----

```

```

|
|
|

```

```

-----
| REPROCESS VERB-TYPE-PROCESS PROCEDURE FROM |
| "IF" CONDITION FALSE STACK-ENTRY (LEVEL |
| POINTER) |
-----

```

```

|
|
|

```

```

-----
DATANAME-REPLACED-OR-VJT

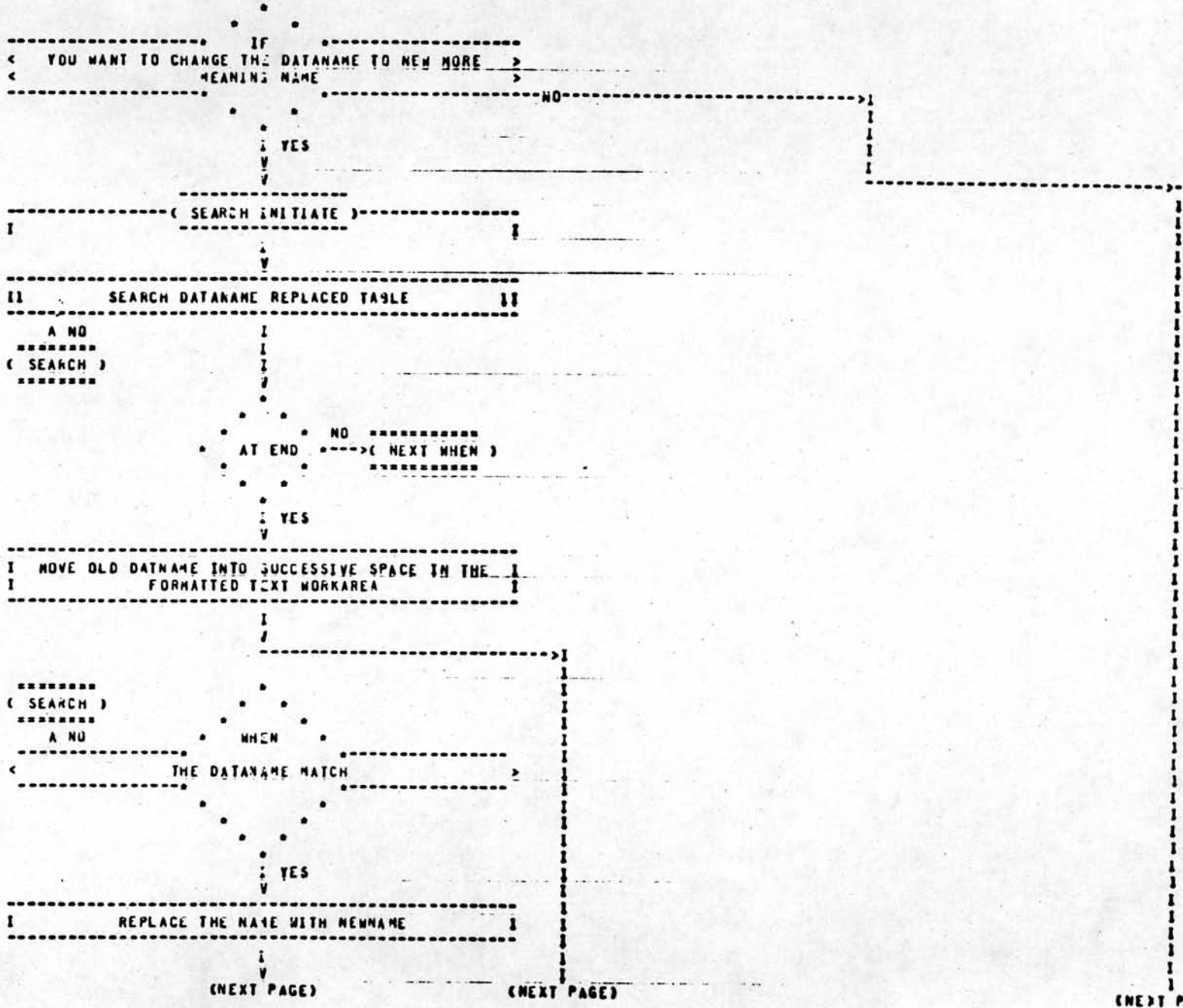
```

```

PROCEDURE # 016
-----

```

(NEXT PAGE)



IF YOU WANT TO CHANGE THE DATANAME TO NEW MORE MEANING NAME

NO

YES

SEARCH INITIATE

SEARCH DATANAME REPLACED TAGLE

A NO
SEARCH

AT END (NEXT WHEN)

YES

MOVE OLD DATNAME INTO SUCCESSIVE SPACE IN THE FORMATTED TEXT WORKAREA

SEARCH
A NO

WHEN THE DATANAME MATCH

YES

REPLACE THE NAME WITH NEWNAME

(NEXT PAGE)

(NEXT PAGE)

(NEXT PAGE)

```

I MOVE IT INTO SUCCESSIVE SPACE IN THE FORMATTED
I TEXT MJKKAREA
I

```

```

.....
PROCEDURE-NUMBER-SEARCH          PROCEDURE # 017
.....

```

```

I ( SEARCH INITIATE ) I
I
I
I SEARCH PROCEDURE DESCRIPTION TABLE II
I
I A NO I
I ( SEARCH ) I
I ( SEARCH ) I
I A NO WHEN I
I ( SEARCH PROCEDURE NAME MATCH ) I
I
I YES I
I
I GIVE ITS NUMBER TO THE PROCEDURE I
I
I
I MOVE THE PROCEDURE NAME AND ITS NUMBER INTO
I SUCCESSIVE SPACE IN THE FORMATTED TEXT
I MOKKAREA
I
I
I

```

```

.....
PROCEDURE-TEXT-PROCESS          PROCEDURE # 010
.....

```

(NEXT PAGE)

WE MUST MANAGE THE ELSE-IF TEXT THAT OCCURS IN THE PREVIOUS PROCEDURE UNTIL NO ELSE-IF TEXTS IN THAT QUEUE. WHEN IT ALSO HAVE ELSE-IF TEXT AGAIN IN ELSE-IF ENTRY OF QUEUE THE ELSE-IF TEXT IS WRITTEN IN THE TAIL ENTRY OF THE QUEUE.

```
I MOVE THE PROCEDURE TEXT TO FORMATTED STATEMENT RECORD I
```

```
I  
V  
I
```

```
IF  
ELSE-IF QUEUE NOT EMPTY
```

NO

```
I YES  
V  
I
```

```
I  
II VARYING QUEUE POINTER FROM 1 INCREMENT BY 1 II  
II UNTIL THE ELSE-IF QUEUE IS EMPTY II  
IV VI  
II PERFORM STRING-TO-FORMATTED-STATEMENT ( P012 II  
II II
```

```
I  
V  
I
```

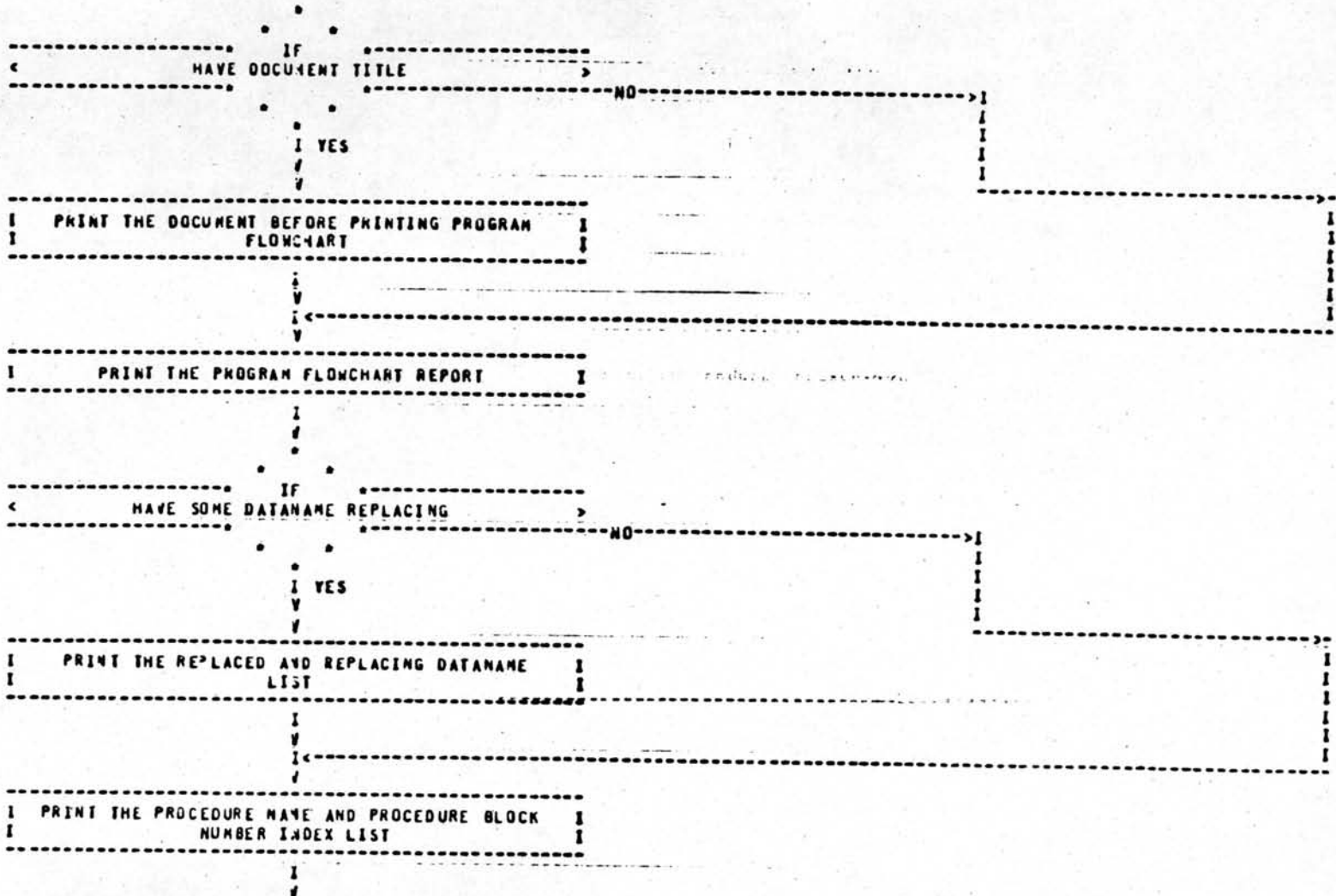
```
/ WRITE THE PROCEDURE TEXT INTO FORMATTED STATEMENT FILE /
```

```
I  
V
```

*****END OF THE THIRD PROCEDURE*****

THE-LAST-PROCEDURE SECTION # 005

- PRINT OUT THE FLOWCHART FROM FORMATTED STATEMENT FILE (NEXT PAGE)



NO.	PROCEDURE NAME	PAGE	NO.	PROCEDURE NAME	PAGE	NO.	PROCEDURE NAME	PAGE
001	MAINLINE	0001	002	THE-FIRST-PROCEDURE	0003	001	VERB-ADDED-PROCEDURE	0004
002	VERB-ADDED-POOL-CREATED	0004	003	VERB-DELETED-PROCEDURE	0005	004	VERB-DELETED-POOL-CREATED	0005
005	DATANAME-REPLACED-PROCEDURE	0006	006	NAME-REPLACED-TABLE-CREATED	0006	007	DOCUMENTATION-TITLE-PROCEDURE	0007
008	DOCUMENT-TABLE-CREATED	0008	003	THE-SECOND-PROCEDURE	0009	009	PROCTAB-AND-NEWSOURCE-CREATED	0009
010	PROCEDURE-TABLE-SETUP	0011	004	THE-THIRD-PROCEDURE	0011	011	TEXT-TO-STRING-CREATED	0013
012	STRING-TO-FORMATTED-STATEMENT	0014	013	VERB-TYPE-PROCESS	0016	014	IF-VERB-PROCESS	0018
015	STACK-DECOMPOSE	0020	016	DATANAME-REPLACED-OR-NOT	0020	017	PROCEDURE-NUMBER-SEARCH	0022
018	PROCEDURE-TEXT-PROCESS	0022	005	THE-LAST-PROCEDURE	0023			

APPENDIX C

EXAMPLE OF COBOL PROGRAM FLOWCHART

THE PROGRAM DETAIL FLOWCHART REPORT
 WITH DATANAME REPLACING FEATURE

FOR THE PROBLEM STATEMENT:-
 UPDATE STUDENT FILE WITH STUDENT TRANSACTION
 CARD FILE AND PRINTING THE TRANSACTION RECORD.

THE COBOL PROGRAM IS

PROCEDURE DIVISION.
 PREPARATION-ROUTINE.

OPEN INPUT CARDFILE M-FILE OUTPUT
 PRINT-FILE N-M-FILE.

HEADING-ROUTINE.

MOVE H TO P-REC WRITE P-REC BEFORE ADVANCING 2.
 MAIN-SEQUENCE. READ CARDFILE AT END MOVE '999999' TO NO.
 READ-OLD-MASTER.

READ M-FILE AT END CLJSE M-FILE N-M-FILE CARDFILE
 PRINT-FILE STOP RUN.

IF NO = KEY THEN MOVE CARDREC TO N-M-REC WRITE N-M-REC
 ELSE MOVE M-REC TO N-M-REC WRITE N-M-REC
 GO TO READ-OLD-MASTER.
 MOVE CARDREC TO P-REC WRITE P-REC AT END-OF-PAGE GO TO
 HEADING-ROUTINE.
 GO TO MAIN-SEQUENCE.

NOW WE WANT TO GENERATE THIS PROGRAM FLOWCHARTING
 BY REPLACING DATANAME IN THE PROGRAM

CARDFILE	----->	STUDENT-TRANSACTION-INPUTFILE
M-FILE	----->	OLD-MASTER-FILE
N-M-FILE	----->	NEW-MASTER-FILE
PRINT-FILE	---->	STONT-TRANSACTION-REPORT
H	----->	HEADING
P-REC	----->	TRANSACTION-LISTING-RECORD
NO	----->	STONT-TRANSACTION-KEY
KEY	----->	STONT-MASTER-KEY
CARDREC	----->	STUDENT-TRANSACTION-INPUT-RECORD
M-REC	----->	OLD-MASTER-RECORD
N-M-REC	----->	NEW-MASTER-RECORD

HERE IS THE PROGRAM FLOWCHART

PREPARATION-ROUTINE

PROCEDURE # 001

```

/ OPEN INPUT STUDENT-TRANSACTION-INPUTFILE /
/ OLD-MASTER-FILE OUTPUT /
/ STDNT-TRANSACTION-REPORT NEW-MASTER-FILE /

```

↓

HEADING-ROUTINE

PROCEDURE # 002

```

I MOVE HEADING TO TRANSACTION-LISTING-RECORD I

```

↓

```

/ WRITE TRANSACTION-LISTING-RECORD BEFORE /
/ ADVANCING 2 /

```

↓

MAIN-SEQUENCE

PROCEDURE # 003

```

/ READ STUDENT-TRANSACTION-INPUTFILE /

```

↓

```

    AT END -----NO----->

```

```

    YES

```

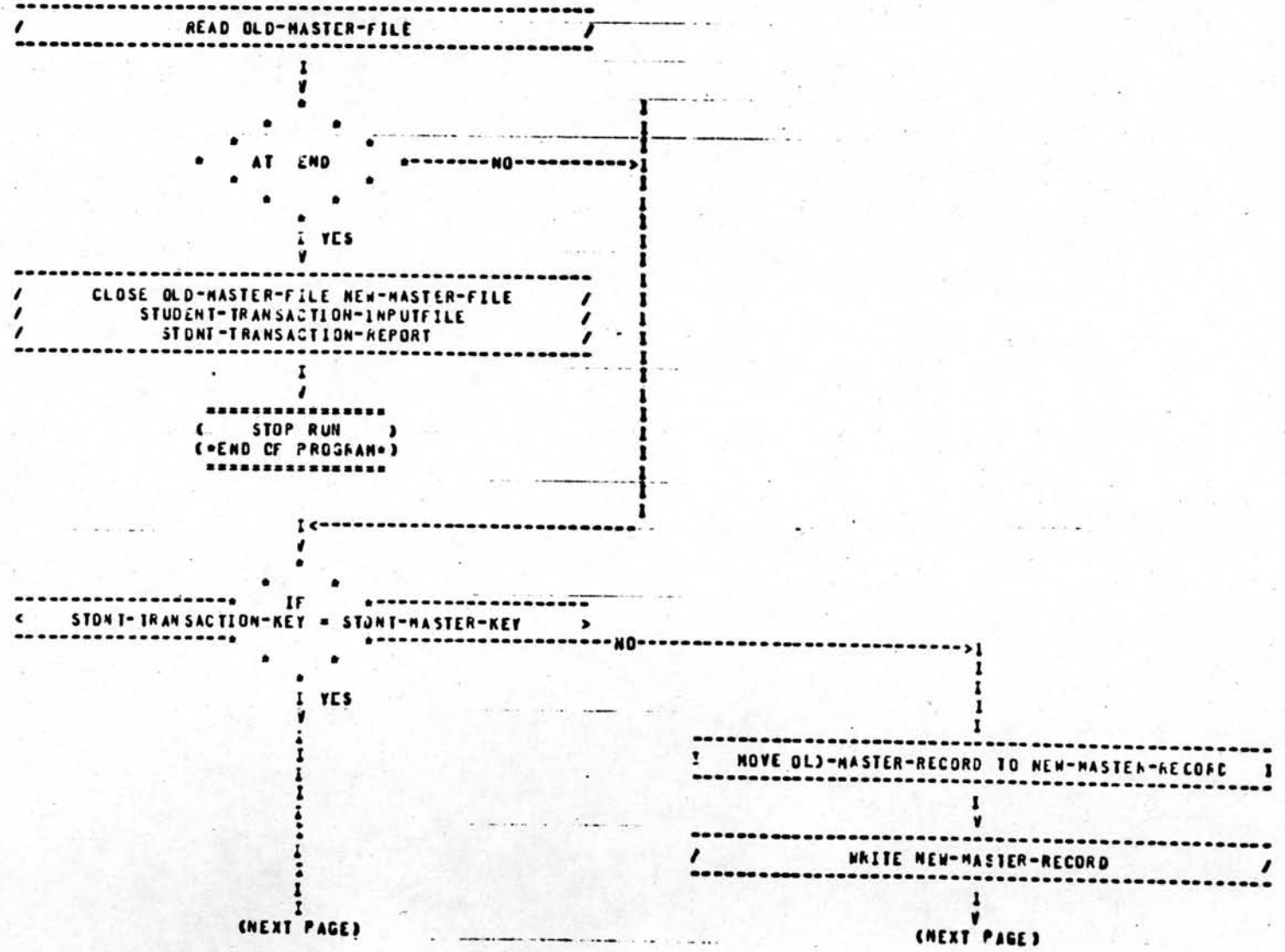
```

I MOVE '999999' TO STDNT-TRANSACTION-KEY I

```

↓

READ-OLD-MASTER PROCEDURE # 004



(PJ04)

I
I
I
I

I MOVE STUDENT-TRANSACTION-INPUT-RECORD TO
NEW-MASTER-RECORD I

I
/

WRITE NEW-MASTER-RECORD /

I
/

I MOVE STUDENT-TRANSACTION-INPUT-RECORD TO
TRANSACTION-LISTING-RECORD I

I
/

WRITE TRANSACTION-LISTING-RECORD /

I
/

• END OF PAGE •
•
•
I YES
/

(PJ02)

I <
/ I

(PJ03)



NO. PROCEDURE NAME
001 PREPARATION-ROUTINE
004 READ-OLD-MASTER

PAGE NO. PROCEDURE NAME
0002 002 HEADING-ROUTINE
0003

PAGE NO. PROCEDURE NAME
0002 003 MAIN-SEQUENCE

PAGE
0002

REPLACED DATA-NAME
CARDFILE
N-M-FILE
H
NO
CARDREC
N-M-REC

REPLACING DATA-NAME
STUDENT-TRANSACTION-INPUTFILE
NEW-MASTER-FILE
HEADING
STONT-TRANSACTION-KEY
STUDENT-TRANSACTION-INPUT-RECD
NEW-MASTER-RECORD

REPLACED DATA-NAME
N-FILE
PRINT-FILE
P-REC
KEY
N-REC

REPLACING DAT-NAME
OLD-MASTER-FILE
STONT-TRANSACTION-REPORT
TRANSACTION-LISTING-RECORD
STONT-MASTER-KEY
OLD-MASTER-RECORD

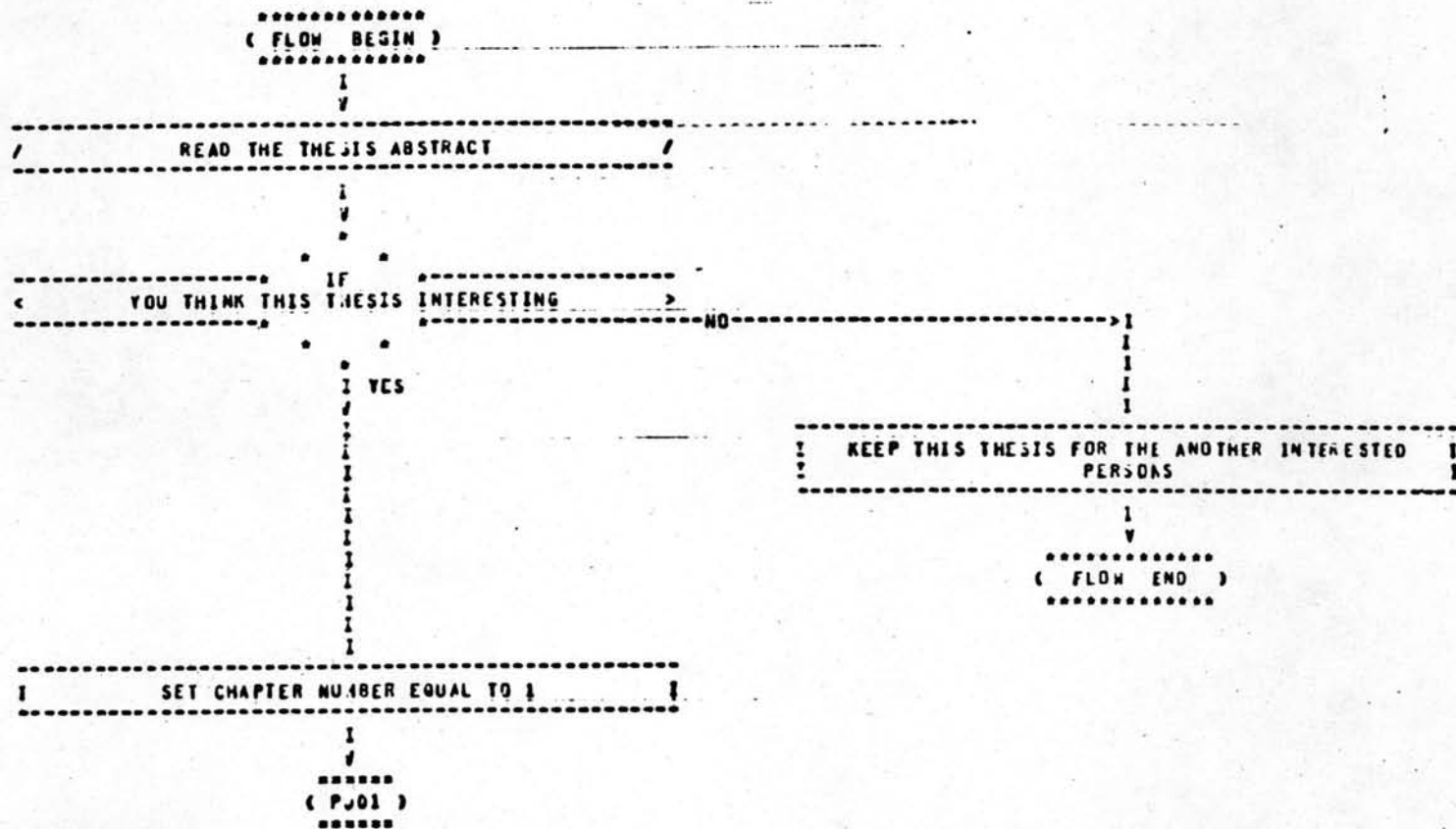
APPENDIX D
PROCEDURE FOR READING THE THESIS



APPENDIX D
PROCEDURE FOR READING THE THESIS

PROCEDURE FOR READING THIS THESIS
 THIS FLOWCHART GENERATED FROM THE FLOWCHART GENERATOR

BEGIN READING THIS PROCEDURE CONTINUE TO FOLLOW THE STEPS



READING-PROCESS

PROCEDURE # 001

1 CHECK TOPICS IN THE CHAPTER 1

IF IT STILL INTERESTING

YES

NO

1 SET VERY MUCH SORE INDEED TO YOU 1

1 PLEASE SAVE IT FOR THE OTHERS INTERESTED PERSON 1

(FLOW END)

/ READ DETAIL IN THIS CHAPTER FROM LINE BY LINE /

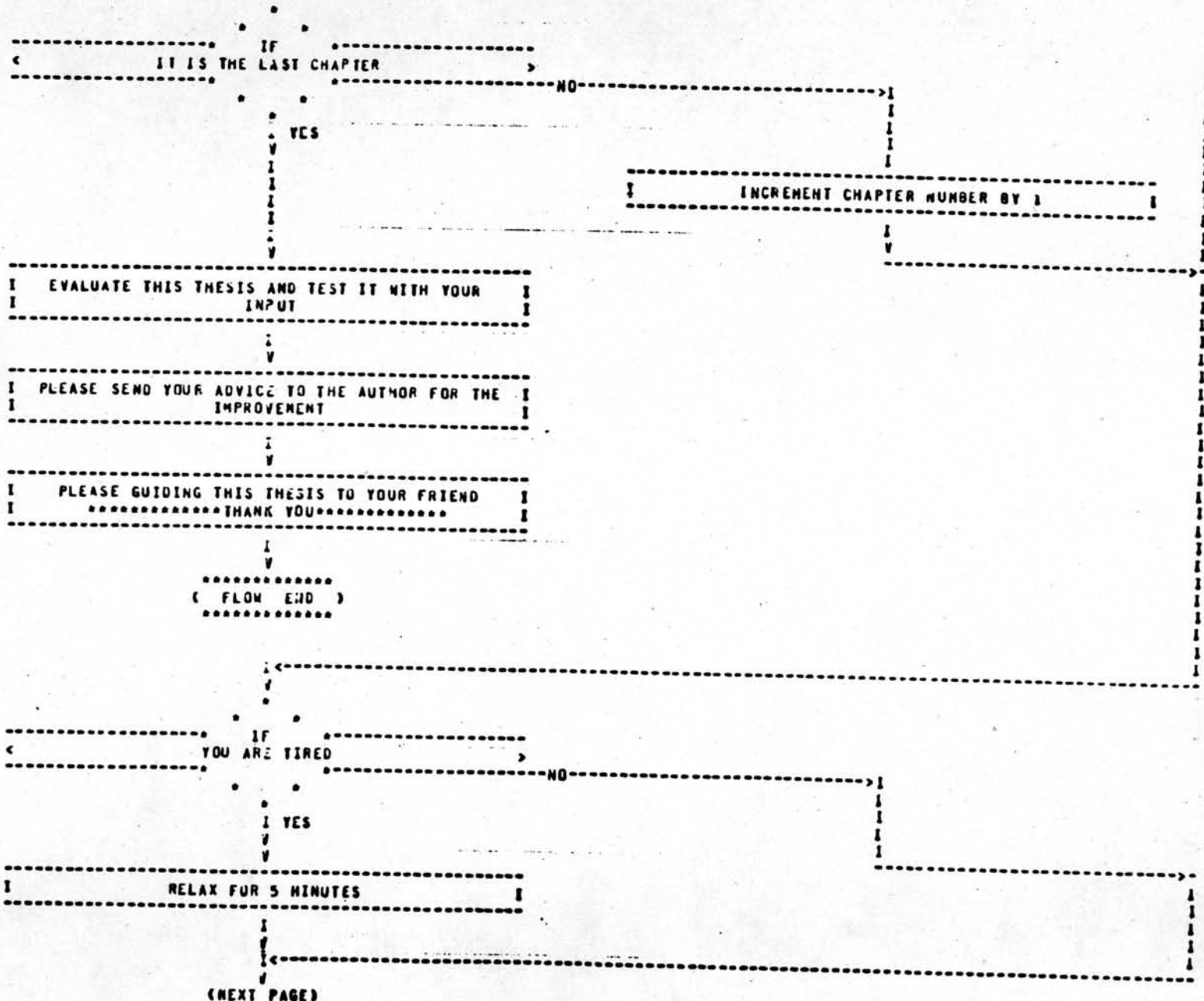
IF IT IS END-OF-CHAPTER

YES

NO

(NEXT PAGE)

(NEXT PAGE)



(FLOW END)

(NEXT PAGE)

(P001)

.....
END-OF-FLOWCHART

.....
PROCEDURE # 002
.....

APPENDIX E.

THE "FLOWCHART GENERATOR" USER INSTRUCTION

IN THIS APPENDIX, THE JOB CONTROL PROCEDURE FOR EXECUTION THE PROGRAM FLOWCHART GENERATOR ARE SHOWN, AS FOLLOW

1) JOB CONTROL CARDS (STATEMENT NO. 001 THRU 011) AT Page e.4)

STMT NO.	LOGICAL UNIT	SYMBOLIC DEVICE	DEVICE TYPE	JOB CONTROL DESCRIPTION
002	SYS001	READER	CARD	ASSIGNMENT CARD FOR READING THE ADDITION REQUIREMENT PARAMETER CARDS
003	SYS002	READER TAPE DISK	CARD TAPE DISK	ASSIGNMENT CARD FOR READING THE SOURCE INPUT THAT WANTED TO BE GENERATED (3 DEVICE TYPES ARE ALLOWED)
004	SYS003	PRINTER	PRINTER	ASSIGNMENT CARD FOR PRINTING THE PROGRAM OR OUTLINE FLOWCHARTING REPORT
005 -007	SYS004	DISK	DISK	ALLOCATED WORKAREA USING IN GENERATING PROCEDURE (STMT PRE-FORMATTING STEP)
008 -010	SYS005	DISK	DISK	ALLOCATED WORKAREA USING IN GENERATING PROCEDURE (TEXT FORMATTING STEP)
011	-	-	-	THE EXECUTION CARD OF THE GENERATING PROGRAM PHASE CATALOGED IN LIBRARY

2) THE ADDITION REQUIREMENT INPUT PARAMETERS (STMNT NO. 012-040)
at page e.4

2.1) THE TITLE CARDS SET

STMNT NO.	FIELD NO.	COLUMN	DESCRIPTION
012	-	08-19	THE TITLE HEADER CARD, IT MUST BE THE FIRST OF TITLE CARDS
013-025	1	02-06	THE TITLE LINE CONTROL FIELD - 'SKIPI' OR BLANK FIELD IS TITLE TEXT WRITTEN CONTROL AFTER ADVANCING 1 LINE - 'SKIP2' ----- AFTER ADVANCING 2 LINES - 'SKIP3' ----- AFTER ADVANCING 3 LINES - 'EJECT' ----- AFTER ADVANCING PAGE - 'FOOT' ----- AT THE END OF PAGE
	2	07	THE CONTINUATION LINE CODE FIELD, IT MUST BE THE CHARACTER 'HYPHEN' (-). IT USED FOR CONCATENATING THE TEXT IN PRECEDE LINE WITH THE TEXT IN CONTINUATION LINE
	3	08-72	TITLE TEXT AREA, CONTAINING TITLE HEAD OF THE FLOWCHARTING REPORT, THE USER CAN GIVE HIS OWN COMMENT AND TITLE OF THE FLOWCHARTING REPORT IN THIS TEXT AREA.
027		08-19	TITLE TRAILER CARD

2.2) THE 'UNAFFECTED FLOW' VERBS DELETED-OUT CARDS SET

STMNT NO.	FIELD NO.	COLUMN	DESCRIPTION
028	-	08-22	DELETED HEADER CARD
029	-	08-72	THE LIST OF DELETED 'FLOW UNAFFECTION' VERBS, AT LEAST ONE BLANK MUST USED AS VERBS SEPARATOR
030	-	08-22	DELETED TRAILER CARD

2.3) THE EXTENDED COBOL VERBS ADDED-IN CARDS SET

STMT NO.	FIELD NO.	COLUMN	DESCRIPTION
031	-	08-22	ADDED VERB HEADER CARD
032	-	08-72	THE LIST OF ADDED COBOL EXTENSION VERBS, AT LEAST ONE BLANK IS USED AS THE VERBS SEPARATOR
033	-	08-22	ADDED VERB TRAILER CARD

2.4) THE NEW MORE MEANING NAME REPLACEMENT CARDS SET

STMT NO.	FIELD NO.	COLUMN	DESCRIPTION
034	-	08-22	'NEW MORE MEANING NAME' HEADER CARD
035-037	-	08-72	THE PAIR OF OLD DATANAME AND NEW MORE MEANING NAME, ONLY ONE PAIR IS APPEARED IN ONE CARD AND ONLY ONE BLANK IS USED AS NAME SEPERATOR
038	-	08-22	'NEW MORE MEANING NAME' TRAILER CARD
039	-	08-22	THE ENDED CARD OF ADDITIONAL REQUIREMENT INPUT

REMARKS THE ORDER SEQUENCE OF THE 4 CARDS SETS IS NOT NECESSARY SAME AS THE EXAMPLE SEQUENCE

3) THE LAST PART OF THESE INPUT SETS IS THE SOURCE PROGRAM OR OUTLINE INPUT THAT WANTED TO BE GENERATED. ONLY 'PROCEDURE DIVISION' PART OF SOURCE PROGRAM IS REQUIRED TO GENERATE, THOUGH THE OTHERS DIVISION ARE INCLUDED, THE GENERATOR PROGRAM NEGLECTED THEM.

1...5...10...15...20...25...30...35...40...45...50...55...60...65...70...75...80 STATEMENT NO.

```
// JOB FLOWGEN      THE PROGRAM FLOWCHART GENERATOR      00001
// ASSIGN SYS001,READER      00002
// ASSIGN SYS002,READER      00003
// ASSIGN SYS005,PRINTER      00004
// ASSIGN SYS003,DISK,VOL=CUWRK1,SHR      00005
// DLBL SYS003,'WORK-FILE1',0      00006
// EXTENT SYS003,CUWRK1,1,0,3600,1200      00007
// ASSIGN SYS004,DISK,VOL=CUWRK1,SHR      00008
// DLBL SYS004,'WORK-FILE2',0      00009
// EXTENT SYS004,CUWRK2,1,0,4800,2400      00010
// EXEC FLOWGEN      00011
      COMMENT LIST      00012
FJFCT      PROGRAM DESCRB      CUSTOMER NAME AND ADDRESS CHAIN      00013
SKIP2      PROGRAM ID      CISNAME      00014
SKIP2      PROGRAM NUMBER      CIS0010      00015
SKIP2      PROGRAM PURPOSE      TO LINK THE CUSTOMER NAME AND ADDRES      00016
      -S      00017
SKIP1      INTO CUSTOMER INFORMATION FILE      00018
SKIP2      FREQUENCY USED      AS REQUIRED      00019
SKIP3      SYSTEM NAME      CIS BANKING DEPOSIT      00020
SKIP2      PROGRAMMER      SATIRA POOLSRUP      00021
SKIP2      ANALYST      SATIRA POOLSRUP      00022
SKIP3      DEVELOPED DATE      10/09/91      00023
SKIP2      MODIFIED DATE      15/09/91      00024
EQ01      NOTE      WHEN SOME PROBLEM OCCURS IN THIS PROCESS, PLEASE CONTACT      00025
      - WITH THE MAINTENACE PROGRAMMER      00026
      COMMENT END      00027
      DELETE-VERB LIST      00028
      TRACE      00029
      DELETE-VERB END      00030
      ADDED-VERB LIST      00031
      PRINT      00032
      ADD-VERB END      00033
      MATCH-NAME LIST      00034
      ST VSAM-FILE-STATUS      00035
      STAT-ACC CUSTOMER-ACCOUNT-KEY      00036
      ACC-NO CUSTOMER-ACCOUNT-NUMBER      00037
      MATCH-NAME END      00038
      END-OF-PROC      00039
```

/*

1...5...10...15...20...25...30...35...40...45...50...55...60...65...70...75...80

PAGE e.5

IDENTIFICATION DIVISION.

PROGRAM-ID. PDP9A090

ENVIRONMENT DIVISION.

CONFIGURATION SECTION.

SOURCE-COMPUTER. IBM-4331.

OBJECT-COMPUTER. IBM-4331.

INPUT-OUTPUT SECTION.

FILE-CONTROL.

SELECT STATEMENT-FILE ASSIGN TO SYS001-04-FR01-SATIRA
ORGANIZATION IS INDEXED
ACCESS MODE IS RANDOM
RECORD KEY IS STAT-ACC
FILE STATUS IS ST.

SELECT NAME-ADDR-FILE ASSIGN TO SYS010-0T-3420-S.

DATA DIVISION.

FILE SECTION.

FD STATEMENT-FILE

LABEL RECORD ARE STANDARD.

01 STATEMENT-REC.

03 STAT-ACC PIC 9(11) COMP-3.

03 CUSTOMER-NAME PIC X(40).

03 CUSTOMER-ADDRESS PIC X(90).

03 FILLER PIC X(54).

FD NAME-ADDR-FILE

LABEL RECORDS ARE OMITTED

RECORD CONTAINS 150 CHARACTERS

BLOCK CONTAINS 10 RECORDS

DATA RECORD IS NAME-ADDR-REC.

01 NAME-ADDR-REC.

03 ACCT-NO PIC 9(11) COMP-3.

03 FILLER PIC X.

03 NAME PIC X(40).

03 ADDR PIC X(90).

03 OTHER-INFORMATION PIC X(23).

WORKING-STORAGE SECTION.

77 ST PIC 99 VALUE 0.

PROCEDURE DIVISION.

MAIN-LINES.

OPEN INPUT NAME-ADDR-FILE. OPEN I-O STATEMENT-FILE.

* ASSUME THE VERB 'TRACE' IS INCLUDED IN SOURCE PROGRAM *
* THE VERB IS DELETED OUT OF THE GENERATED FLOWCHART *

TRACE ON

IF ST NOT = 0 THEN DISPLAY 'STATEMENT FILE NOT EXIST'
OPEN OUTPUT STATEMENT-FILE IF ST NOT = 0 THEN
DISPLAY 'OPEN STATEMENT FILE ERROR, VSAM CODE = '
ST STOP RUN.
TRACE OFF

* ASSUME VERB 'PRINT' IS INCLUDE IN SOURCE PROGRAM *

----- THE 'PRINT' VERB IS ALSO GENERATED WITHIN THE FLOWCHART

READ-NAME-ADDRESS. READ NAME-ADDR-FILE AT END
PRINT STATEMENT-FILE CLOSE NAME-ADDR-FILE STATEMENT-FILE
STOP RUN.

MOVE ACCT-NO TO STAT-ACC.
MOVE NAME TO CUSTOMER-NAME MOVE ADDR TO CUSTOMER-ADDRESS.
READ STATEMENT-FILE.
IF ST = 0 THEN REWRITE STATEMENT-REC
INVALID KEY DISPLAY 'REWRITE ERROR VSAM CODE = '
ST 'ACCOUNT KEY = ' STAT-ACC
STOP RUN.
IF ST NOT = 0 THEN DISPLAY STAT-ACC ' ----- NEW CUSTOMER
' ACCOUNT NUMBER' WRITE STATEMENT-REC
INVALID KEY DISPLAY 'VSAM WRITTEN ERROR KEY = '
STAT-ACC STOP RUN.
GO TO READ-NAME-ADDRESS.

/*
/G

PROGRAM NAME CUSTOMER NAME AND ADDRESS CHAIN
PROGRAM ID CISNAME
PROGRAM NUMBER CIS0010
PROGRAM PURPOSE TO LINK THE CUSTOMER NAME AND ADDRESS
 INTO CUSTOMER INFORMATION FILE
FREQUENCY USED AS REQUIRED

SYSTEM NAME CIS BANKING DEPOSIT
PROGRAMMER SATIRA POOLSRUP
ANALYST SATIRA POOLSRUP

DEVELOPED DATE 10/09/91
MODIFIED DATE 15/09/91

NOTE WHEN SOME PROBLEM OCCURS IN THIS PROCESS, PLEASE CONTACT WITH THE MAINTENANCE PROGRAMMER

MAIN-LINES

PROCEDURE # 001

/ OPEN INPUT NAME-ADDR-FILE /

I
V

/ OPEN I-O STATEMENT-FILE /

I
V

* * *
* IF *
* VSAM-FILE-STATUS NOT = 0 *
* * *

I YES
V
V

/ DISPLAY 'STATEMENT FILE NOT EXIST' /

I
V

/ OPEN OUTPUT STATEMENT-FILE /

I
V

* * *
* IF *
* VSAM-FILE-STATUS NOT = 0 *
* * *

I YES
V
V

/ DISPLAY 'OPEN STATEMENT FILE ERROR VSAM CODE
/ = ' VSAM-FILE-STATUS /

I
V

(NEXT PAGE)

I
I
I
I

I
I
I
I

(NEXT PAGE)

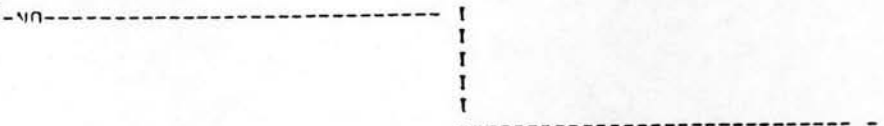
I MOVE ADDR TO CUSTOMER-ADDRESS I

I
V

/ READ STATEMENT-FILE /

I
V
*

< * IF * >
VSAM-FILE-STATUS = 0

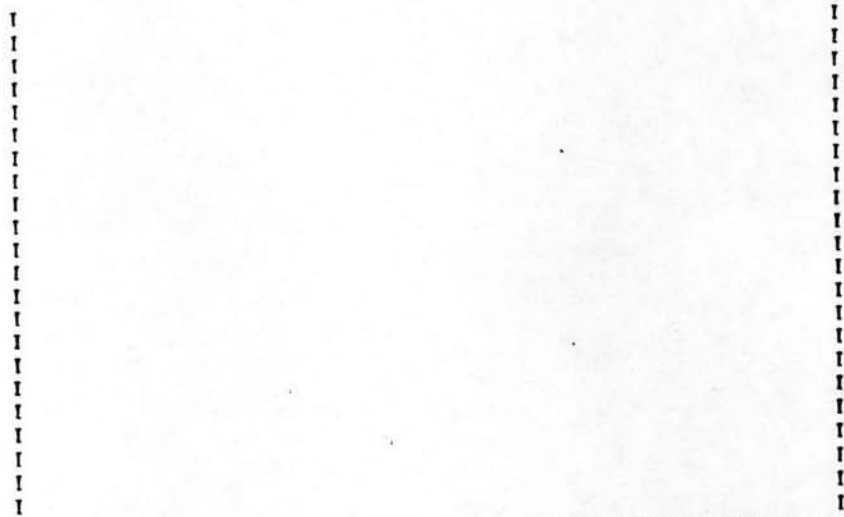


I YES
V
V

/ REWRITE STATEMENT-REC /

I
V
*

* INVALID KEY * <-----NO----->



I YES
V

/ DISPLAY 'REWRITE ERROR VSAM CODE = '
/ VSAM-FILE-STATUS 'ACCOUNT KEY = '
/ CUSTOMER-ACCOUNT-KEY /

I
V

=====
{ STOP R'UN }
{ *END OF PROGRAM* }
=====

I<
V
(NEXT PAGE)

```

      *
      *   *
      *   * IF *
      *   * *
    <----- VSAM-FILE-STATUS NOT = 0 ----->
      *   * *
  
```

NO

```

      I
      V
      V
  
```

```

-----
/  DISPLAY CUSTOMER-ACCOUNT-KEY =   NEW   /
/  CUSTOMER* ACCOUNT NUMBER*         /
-----
  
```

```

      I
      V
  
```

```

-----
/  WRITE STATEMENT-RFC              /
-----
  
```

```

      I
      V
  
```

```

      *
      *   *
      * * INVALID KEY *
      * * *
  
```

NO

```

      I
      V
  
```

```

-----
/  DISPLAY 'VSAM WRITTEN ERROR KEY = '  /
/  CUSTOMER-ACCOUNT-KEY                 /
-----
  
```

```

      I
      V
  
```

```

=====
(  STOP RUN  )
(*END OF PROGRAM*)
=====
  
```

```

I
I
I
I
I
I
I
I
I
I
I
I
I
I
I
  
```

```

I<
V
  
```

```

=====
( P00? )
=====
  
```


NO. PROCEDURE NAME
001 MAIN-LINES

PAGE NO. PROCEDURE NAME
0008 002 READ-NAME-ADDRESS

PAGE NO.
0009 --

PROCEDURE NAME

PAGE

APPENDIX F

DATANAME SIZE AND ENTRY DEFINED IN PROGRAM

DESCRIPTION OF

 DATANAME ENTRIES AND TABLES DEFINED IN PROGRAM

DATANAME DESCRIPTION	DATANAME REFERRED IN PROG.	AT PAGE	DEFINED SIZE/ ENTRY	ENTRY/ TABLE	MAXIMUM
REPLACED-REPLACING DATANAME TABLE	OLD-NEWNAME- TABLE	4.9	65	10	500
NEW VERBS ADDITIONAL	ADDED-VERBS	4.9	10	90	3000
UNAFFECTED FLOW VERBS DELETION	DELETED-VERBS	4.9	10	20	3000
PARA/SECTION DESCRPT. TABLE	PARA-SECTION- TABLE	4.9	38	500	800
NO. OF CHARS/SENTENCE	STRING-CHAR	4.9	10000	-	30000
NO. OF CHARS/STMENT	STATEMENT- BLOCK	4.9	506	-	30000
NO. OF IF VERBS/ SENTENCE	DATANAMES AT ST. NO 431-450	4.10	-	20	1*
NO. OF CHARS/ELSE-IF SENTENCE	ELSE-IF-CHAR	4.9	3000	-	30000
NO. OF ELSE-IF ENTRIES /ONE PARAGRAPH	ELSE-IF-ENTRY	4.10	-	10	2*
NO. OF TITLE LINES	COMMENT-ENTRY	4.10	130	100	250

REMARK 1* MAXIMUM NO. OF ENTRIES X NO. OF CHARS/SENTENCE < 30000
 2* MAXIMUM NO. OF ENTRIES X NO. OF CHARS/ELSE-IF SENTENCE < 30000

BIBIOGRAPHY



NAME MR. SATIRA POOLSRUP

EDUCATION B.A. (Statistics), 1st class hon.,
THAMMASAT UNIVERSITY, 1976

EXPERIENCE SYSTEM PROGRAMMER,
E.D.P. DEPT., BANGKOK BANK LTD., 1976-1978

TECHNICAL REPRESENTATIVE,
BURROUGHS DEPT., YIP IN TSOI LTD., 1979-1980

PROJECT LEADER,
E.D.P. DEPT., SIAM COMMERCIAL BANK LTD., 1981-now