Lib Sc. 13; 1976; PAPER J.

Computerisation of POPSI.

(Subject heading series. 7). (Non-conventional methods in document retrieval. 22).

I K Ravichandra Rao,* *Documentation Research and Training Centre, Indian Statistical Institute, Bangalore* 560003.

[Computer program for generation of alphabetical subject index—Postulate-based Permuted Subject Indexing (=POPSI) is described. Two rules for choice of Lead Term for facets in case of *Parts of Core Entity of Study* are given.]

01 Introduction

In this paper, a study of the procedures pertaining to computerisation of Postulate-based Permuted Subject Indexing (=POPSI) is reported.

Computerisation of subject indexing at DRTC was first begun in 1970–71. This resulted in the successful completion of a research project in 1972. The result of this research project was published in 1973 (4) and then in 1975 (5). Since then, continuous research in this line of thinking has been going on.

02 Definition

1 *Facet*: Facet, as defined by S R Ranganathan, is a generic term used to denote any component—be it a basic subject or an isolate of a compound subject and also its respective, ranked forms, terms and numbers (3).

2 *Descriptor*: A descriptor is a facet, as defined above or a facet with all its levels or a facet with speciator of kind 2. For example,

1 Stimulation;
2 Bacteria, Cell, Wall
3 1976=June=15

1 Scope of the Paper

The paper presents a computer program developed for generation of POPSI. It emphasises the design, operation and implementation of the program from the view point of the designer or systems analyst and manager of information services. Also it provides some idea about the characteristics, capabilities and limitations of the program package.

2 Data Base

21 BIBLIOGRAPHICAL DESCRIPTION

The rules in the *Classified Catalogue Code*, Ed 5 (CCC) and/or *Indian Standard for Bibliographical Reference* (IS: 2381–1963) may be used in cataloguing the documents. The Class Number and the corresponding Feature Heading may be written as given in the specimen entry below:

W
Political Science., Non violence: Evaluation (by) Sikh-Akali religion
OPEN LETTER to Akalis. 1924. (Young India. 1924. Feb. 28).

Y
Sociology, South Africa—European—Commercial class; Class hatred—(towards) Indian national.
ANTI-INDIAN CAMPAIGN in South Africa. 1924. (Young India. 1924 Feb. 21).

A full printed catalogue of bibliographical records may be maintained in a book form with supplement at regular intervals. In addition, a number of special purpose indexes to data fields such as author, subject, series, etc., in Machine Readable Bibliographical Database (=MRBD) may be produced. The detailed description of the preparation of input for and creation of MRBD on magnetic tape using the program CORC has been reported earlier (6).

3 Alphabetical Index to Subjects

An alphabetical index to the subjects of the documents for which entries are made in MRBD may be necessary for a variety of purposes. For example:

1 Formulation of subject headings, which may be used as Feature Headings or for other indexing purposes;
2 Deriving subject index entries for a classified catalogue, an index to a book, etc;
3 Determining the subject of reader's query in a consistent and helpful way; and
4 Formulating a strategy for searching information about a subject in a catalogue or other surrogate files;

31 SUBJECT INDEX ENTRIES

Subject index entries are to be derived from the full subject representation in the Main Entry, Feature Heading Section, such that each of the approach terms selected occurs as lead term of the entries. Display of the subject index entry consists of three sections as follows :

1 Lead Term Section ;
2 Subject Representation Section (Short Display) ; and
3 Reference Section.

*Present address: School of Library and Information Science, University of Western Ontario, London, Ont., Canada.

Rules for choice of Lead Term and rendering in the above-said three sections are published elsewhere (3). It may be noted that this work of A Neelameghan and M A Gopinath is the result of discussion held during DRTC Seminar on *Subject Indexing* in early 1975 (1, 2, 3). But, herein, a further modification has been made in the rules for the choice of Lead Term in case of Parts of Core Entity of Study, as given in Sec 613 of the earlier paper (3). The modifications are:

1 If the Lead Term denotes a part or organ of a Core Entity (as recognised in subject structure), then the term representing all immediate preceding higher level of the concept, should be suffixed to the Lead Term, provided the higher level concept is represented in the short display of the subject.

2 If the preceding facet in Lead Term section is a level of higher order of any facet, then all the immediate preceding levels should be suffixed to it.

This principle is to be followed even if the term denoting a part of a Core Entity is not a Lead Term but occurs as a component of the set of terms in the Lead Term Section. For example:

Microbiology, Bacteria, Cell, Wall, Membrane: Decomposition—Antibiotics

Here, the Lead Terms are

Antibiotics, Decomposition, Membrane, Wall, Cell, Bacteria, Microbiology
Bacteria, Microbiology
Cell, Bacteria, Microbiology
Decomposition, Membrane. Wall, Cell, Bacteria, Microbiology
Membrane, Wall, Cell, Bacteria, Microbiology
Wall, Cell, Bacteria, Microbiology

There is provision for omission of the Basic Subject term Microbiology in the Lead Term section under specific conditions.

4 Program Description

Name of the Program: Postulate-based Permuted Subject Indexing
Acronym: POPSI
Author: I K Ravichandra Rao
Date of Current Version: 1976-02-20.

41 Purpose

The purpose of this program is to generate subject index entries according to Rules for Choice of Lead Term and Rules for Rendering POPSI entries as given in Sec 31 and in the article by Neelameghan and Gopinath (3).

42 Input

1 The input to the program consists of MRBD on magnetic tape with standard label according to COBOL syntax. Identification of magnetic tape is CORC 1975 FILE (7).

2 Also, an input in the form of Control Card is necessary. Table 1 illustrates the Control Card. Control Card provides information regarding the indicator digits used in Feature Heading Section. This information is used to identify the last character of a descriptor or kernel term in Feature Heading Section. These characters may be comma, semi-colon, colon, single inverted comma, dot, double inverted comma, asterisk, plus, comma, hyphen, forward arrow, backward arrow.

Table 1. Card Design of Control Card

SN	No of character	Character position	Description	Remarks
1	1	11	Punch 2	Indicates :
2	1	12	Punch 1	Indicates ;
3	1	14	Punch 5	Indicates =
4	1	15	Punch 4	Indicates >
5	1	17	Punch 8	Indicates
6	1	19	Punch 2	Indicates „
7	1	20	Punch 6	Indicates #
8	1	23	Punch 2	Indicates &
9	1	24	Punch 1	Indicates '
10	1	27	Punch 2	Indicaste *
11	1	28	Punch 2	Indicates +
12	1	29	Punch 1	Indicates ,
13	1	30	Punch 3	Indicates —
14	1	31	Punch 1	Indicates .
15	1	63	Punch 2	Indicates '
16	1	64	Punch 2	Indicates /

Note.—Character position for any indicator digit is equivalent to the ordinal value of that indicator digit *plus* one

43 Output

The output file from POPSI is a magnetic tape, labelled POPSI FILE. While generating subject index entries, that is, POPSI entries, output may be obtained on on-line printer, so that programmers can de-bug the system, if necessary.

44 Procedure

1 A control card is read and information is stored in the working storage area called Work-Table.

2 A bibliographical record is read from CORC 1975 FILE into the working storage area called CORC-ENTRY.

3 Using the relevant tag the address of the first character in Feature Heading Section is obtained. The required address is available in the area called TAG (2, 7).

4 The next step is to construct short display. Short display consists of short form of Subject Representation and Reference Section, *i e*, serial number of bibliographical record. It is stored in the area called SHORT-DISP.

5 A digit called *less than* sign (<) is introduced in Feature Heading Section to indicate higher order levels of facets. For example, the higher order levels of facets may appear as follows:

1 Microbiology, Bacteria, Cell, Wall; Structure
2 Horticulture, Flower, Colour; Variation; Degree: Study

Now, by introducing the digit < in the Feature Heading Section, the subject strings obtained are

1 Microbiology, Bacteria < Cell < Wall; Structure
2 Horticulture, Flower; Colour < Variation < Degree: Study

6 A table of address of descriptors of Lead Term Section is constructed. Lead Table consists of 60 lists. Each list consists of 3 fields. Length of each field is 3 characters. These fields are used for of storing address of Lead Term. If the Lead Term is a speciator then the other two descriptors are Term is principal facet and preceding facets. For example, consider the following three subjects:

1 Microbiology
2 Bacteria; Structure
3 Bacteria: Decomposition—Antibiotics

Now, the Lead Table will have the address of descriptors as given below:

	Field 1	Field 2	Field 3	
For 1	Address of Microbiology	Zeroes	Zeroes	List 1
	Field 1	Field 2	Field 3	
For 2	Address of Bacteria	Zeroes	Zeroes	List 1
	Field 1	Field 2	Field 3	
	Address of Structure	Address of Bacteria	Zeroes	List 2
	Field 1	Field 2	Field 3	
For 3	Address of Bacteria	Zeroes	Zeroes	List 1
	1	2	3	
	Address of Decomposition	Address of Bacteria	Zeroes	List 2
	1	2	3	
	Address of Antibiotics	Address of Decomposition	Address of Bacteria	List 3

Field 1 is used for storing the address of Lead Term. Field 2 is used to store the address of preceding facet or principal facet as the case may be. Field 3 is used to store the preceding facet, when Lead Term is a speciator, otherwise Field 3 contains Zeroes. It may be noted that addresses are stored in binary form. Since Lead Table contains 60 lists, a maximum of 60 POPSI entries can be generated from a single Feature Heading.

7 After constructing Lead Table, using the address of descriptors such as Lead Term, preceding facet and principal facet, Lead Term Section is generated. For the purpose, using this address available in Field 1, a descriptor is brought into the area called FIRST DESCRIPTORS. Similarly, using the addresses available in Field 2 and Field 3, descriptors are brought into the areas called SECOND DESCRIPTORS and THIRD DESCRIPTORS respectively. If the second and third descriptors contain < or = digit then descriptors are inverted and stored in SECOND DESCRIPTORS and THIRD DESCRIPTORS as the case may be. For example, a descriptor may be as follows:

1 Bacteria < Cell < Wall; or
2 1976 = June = 15

Now, inverted descriptors will be

For 1, Wall, Cell, Bacteria
For 2, 15 June 1976

8 A facet from the area called FIRST-DESCRIPTORS is brought into another area called BUFFA. Then descriptors from SECOND DESCRIPTORS and THIRD DESCRIPTORS are attached to the facet and stored in two different areas called BUFFB and BUFFC.

9 Lead Term Section from the area BUFFC along with short display from the area SHORT-DISP is written on to magnetic tape called POPSI FILE.

10 The next facet from BUFFA, if any, is moved into BUFFB and all facets of BUFFC into BUFFB and then the subject string from BUFFB to BUFFC. Step 9 is repeated with all the facets in BUFFA. For example, consider the following case: FIRST-DESCRIPTOR may have either

1 Horticulture; or
2 Colour < Depth < Variation

SECOND DESCRIPTOR has the descriptor Study and THIRD DESCRIPTOR contains only *spaces*. Now, steps 8, 9 and 10 will generate the following Lead Sections:

For 1 Horticulture, Study
For 2—Cycle 1: Colour, Study
For 2—Cycle 2: Variation, Colour, Study
For 2—Cycle 3: Variation, Depth, Colour, Study

11 Steps 7, 8, 9 and 10 are repeated for all lists in Lead Table. However, if all three fields in a list consist of zeroes, step 11 is terminal step.

12 Steps 2 to 11 are repeated for all bibliographical records of CORC 1975 FILE.

13 Using the information as given in Sec 7, is sorted POPSI FILE.

14 Printout of POPSI FILE may be obtained, if necessary.

5 Some Steps in Programming

51 CONSTRUCTION OF SHORT DISPLAY
1 Let i, j and k be equal to 1
2 Extract the i^{th} character from Feature Heading Section into an area called X_0
3 If X0 = " # " then go to step 24
4 If X0 = " [" then go to step 18
5 If X0 = " < " then go to step 14
6 If X0 = " / " then go to step 13
7 Move i^{th} character from Feature Heading Section to j^{th} character position in SHORT DISP
8 If X0 = comma, or semi-colon, or dot, or single inverted comma or colon, or asterisk or double inverted comma, or backward arrow, or ampersand, or plus, or upward arrow, or hyphen, or equal sign, then go to step 16
9 i ← i + 1
10 j ← j + 1
11 i ← 1
12 If j > 210 then go to step 24 else go to step 2
13 i ← i + 1; go to step 2
14 j ← k; 1 ← 2
15 j ← j + 1; go to step 17
16 k ← j
17 i ← i + 1, If i is equal to 1 then go to step 10 otherwise go to step 11
18 i ← i + 1
19 Extract the i^{th} character from Feature Heading into X0
20 If X0 = " # " then go to step 23
21 If X0 = "] " then go to step 22 else go to step 18.
22 Increase the value of i by 1 till the i^{th} character in the Feature Heading Section is a non-space or non-indicator digit and then go to step 11.
23 Display " *missing record omitted*". Go to step 1 after reading next bibliographical entry from tape].
24 Move *space* to i^{th} character position of SHORTDISP.
25 i ← i + 1, repeat step 24 once again
26 Repeat steps 24 and 25 once again
27 j ← 1
28 Move j^{th} character of Serial Number to i^{th} character position of SHORTDISP
29 i ← i + 1; j ← j + 1
30 Repeat step 28 and 29 four times.
31 i ← i + 1. Move *space* to i^{th} character position of SHORT-DISP till i > 212
32 Short-display construction is over; Pause.

52 INTRODUCING *less than* SIGN INTO FEATURE HEADING SECTION
1 Let i be equal to 1; and k = zero
2 Extract the i^{th} character from Feature Heading Section into X0
3 If X0 = " # " then go to step 11
4 If X0 = comma, semicolon, dot, single inverted comma, go to step 5 else go to step 7
5 If X0 = k, go to step 6, else go to step 9
6 Move *less than* sign to i^{th} character position of Feature Heading Section. Go to step 10
7 If X0 = " : " then k ← Zero
8 Go to step 10
9 k ← X0
10 i ← i + 1; go to step 2
11 Pause [Levels have been removed. Here it may be noted that a proper merging of Sec 51 and Sec 52 will save computer time required for the purpose of generation of subject index entries].

53 CONSTRUCTION of LEAD TABLE
1 Move all zeroes to Lead Table
Note: Lead Table has 60 lists, called DESC-LOC (1), DESC-LOC (2) DESC-LOC (60).
Each list has 3 fields called DA DB and DC.
2 Move zeroes to XDA, XDB, XDC, BITA, BITB and $X0_0$←Tag (2)
3 If BITB = 1 then go to Step 23
4 X0 → DA of DESC—LOC (XD)
5 XDB → DB of DESC—LOC (XD)
6 XDC → DC of DESC—LOC (XD)
7 If BITA ≠ zero then go to Step 9 else DXDB ← XDB
8 XDB ← X0
9 X_0 ← X_0+1
10 If CS (X0) = "#" (if the character to be processed is hash mark), then go to step 32
11 If CS (X0) = *Indicator digit* then go to step 14
12 If CS (X0) = *hyphen* then go to step 20
13 If CS (X0) = *Greater than* sign then go to step 22 else go to step 9
14 XDC, BITA, BITB ← 0
15 X0 ← X0+1
16 If CS (X0) = *Space* then go to step 15
17 If XD > 58 then go to step 32
18 XD ← XD+1
19 Go to step 3
20 BITA ← 1
21 XDC ← DXDB; go to step 15
22 If BITA ≠ 1 then BITB ← 1; go to step 1
23 If BITA ≠ zero then go to step 30
24 XDC ← zero
25 XDB ← DXDB
26 Repeat step 4, step 5 and step 6
27 DXDB ← XDB
28 XDB ← X0
29 Go to step 9
30 Repeat step 4, step 5, and step 6
31 Go to step 9
32 Lead Table construction over; Pause

54 CONSTRUCTION OF LEAD TERM SECTION

Note: 1 Define the following area in the working storage sections. For the purpose of convenience, COBOL statements are used to define the necessary area.

```
01 FIRST—DESCRIPTORS
   02 FID OCCURS 120 TIMES
   PIC X.
01 SECOND-DESCRIPTORS
   02 SED OCCURS 120 TIMES
   PIC X.
01 THIRD DESCRIPTORS
   02 THIS OCCURS 120 TIMES
     PIC X.
01 BUFFA.
   02 BF OCCURS 120 TIMES PIC X.
01 BUFFB.
   02 BB OCCURS 120 TIMES PIC X.
01 BUFFC.
   02 BC OCCURS 120 TIMES PIC X.
```

2 X0, X1, X2, X3, X4, X5, X6, XL, XS, are the locations with the PIC 1 (24).

541 *Get the Descriptors into FIRST-DESCRIPTORS, SECOND-DESCRIPTORS and THIRD-DESCRIPTORS*

1 XD, X1→ 1
2 Move all spaces to BUFFC
3 BITA ← 0; X0 ← DA of DESC-LOC (XD)
4 If X0 = Zero then read next bibliographical record from tape and then go to sec 51
5 Repeat step 1 thru step 16 of Sec 541
6 FIRST-DESCRIPTORS ← BUFFA;
7 X0 ← DB of DESC-LOC (XD); X1 — 1; If X0 = Zero then go to step 16
8 Repeat step 1 thru step 16 of Sec 541
9 If BITA = Zero then go to step 10 else repeat Sec 543
10 SECOND-DESCRIPTORS ← BUFFA; BITA ← Zero
11 X0 ← DC of DESC-LOC (XD)
12 X1 ← 1; If X0 = Zero then go to step 17 else
13 Repeat step 1 thru step 16 of Sec 541
14 If BITA = Zero then go to step 16 else repeat Sec 543
15 THIRD-DESCRIPTORS ← BUFFA; XS, XL ← 1; Go to Sec 544
16 SED (X1) ← "≠"
17 THD (X1) ← "#", XS, XL ← 1; Go to Sec 544

542 *Get the Descriptor into BUFFA*

Note: CS (1), CS (2), ··· refer to characters in Feature Heading Section.

1 Move all the spaces to BUFFA
2 If CS (X0) = "(" then go to step 10.
3 If CS (X0) = "(" or ")" then $X_0 \leftarrow X_0 + 1$
4 If CS (X0) = any indicator digit or Hash mark then go to step 16
5 BA (X1) ← CS (X_0)
6 If CS (X_0) = "<" or "=" then BITA —
7 X0 ← X0 + 1
8 X1 ← X1 + 1
9 Go to step 2
10 X0 ← X0 + 1
11 If CS (X0) = "#" then go to step 15
12 CS (X0) = ")" then go to step 13 else go to step 10
13 X0 ← X0 + 1; Increase X_0 till CS (X0) is a non-space character
14 Go to step 3
15 Display "*missing, record omitted*" and then go to step 1 of Sec 51 after reading bibliographical record from the data base
16 Descriptor is now available in BUFFA; Move "#" into BA (X1); Pause.

543 *Reversing Descriptors,*

1 Move all spaces to BUFFB, BUFFC
2 X5, X6 ← zero
3 X1, X2, X3 ← 1
4 BC (X3) ← "#"
5 BITB ← 1
6 If BA (X1) = *Space*, then go to step 14
7 If BA (X1) = "#", then go to step 16
8 If BA (X1) = "<", then go to step 19
9 If BA (X1) = "=", then go to step 21
10 BB (X2) ← BA (X1)
11 X1 ← X1 + 1
12 X2 ← X2 + 1
13 Go to step 7
14 X1 ← X1 + 1
15 Go to step 6
16 If X6 = 2, then BB (X2) space else BB (X2) ','
17 X5 ← 0
18 Go to step 23
19 X6 ← 1
20 Go to step 22
21 X6 ← 2
22 X5 ← 1
23 If BC (1) = "#" then go to step 28 else if X6 = 2 then BB (X2) ← *space* else B B (X2) ← ','
24 X2 ← X2 + 1; BB (X2) ← BC (X3)
25 If BC (X3) = "#" then go to step 29
26 X3 ← X3 + 1
27 Go to step 24
28 BB (X2) ← BC (1)
29 BUFFC ← BUFFB
30 BB (X2) ← space
31 If X5 = zero then go to step 37
32 X1 ← X1 + 1
33 X1 ← X1 + 1
34 If BA (X1) = *space*, then go to step 33
35 X2, X3 ← 1
36 If BITB = 1 then go to step 5 else go to step of Sec 544
37 BUFFA ← BUFFC

544 *Construction of Lead Term Section*

```
1  XDA ← zeroes
2  If FID (1) = " / " then go to step 17
3  If SED (XS) = " # "
4  BC (XC) ← SED (XS)
5  XL ← XL + 1
6  XS ← XS + 1
7  Go to step 3
8  XS ← 1
9  If XL = 1, then go to step 12
10 If THD (XS) = " # " then go to step 26
11 XL ← XL + 1
12 If THD (XS) = " # ", then go to step 26
13 BC (XL) ← THD (XS)
14 XL ← XL + 1
15 XS ← XS + 1
16 Go to step 12
17 FID (1) ← space
18 i ← 1
19 BC (XL) ← i^th character in " sec "
20 XL ← XL + 1
21 i ← i + 1
22 If XL ← 7 then go to step 23 else go to step 19
23 XS ← 1
24 XDA ← 1
25 Go to step 3
26 BC (XL) ← " # "
27 BUFFA ← FIRST—DESCRIPTORS
28 BUFFB ← spaces
29 X6, X5 ← zeroes
30 X1, X2, X3 ← 1
31 BITB ← 2
32 XDC ← 1
33 Repeat step 6 to step 37 of Sec 543
34 XDC ← 2
35 LEAD-SEC → BUFFB
36 If XDA = zero, then go to step 37 else go
   to step 43
37 Write POPSI entry on tape
38 BUFFB ← spaces
39 If XDC = 1, then go to step 32, else
   go to step 40
40 XD ← XD + 1
41 X1 ← 1
42 Go to step 2 of Sec 541
43 SHORT-DISPLAY ← SHORT-DISP
44 SHORT-DISP ← spaces
45 Write POPSI entry
46 SHORT-DISP ← SHORT-DISPLAY
47 Go to step 38
```

6 Sample Input-Output

Sec 93 gives a sample input used for the purpose of generating subject index entries and Sec 94 gives the sample output.

7 Operating Procedure

1 Using the Program CORC, Machine Readable Bibliographical Data Base is to be created (7).

2 The program POPSI must be compiled, loaded and executed. After the compilation, object program may be dumped on punch card. Then, next time when the program is run, the compilation step may be omitted and the binary file, which is on cards, may be directly loaded and executed.

3 POPSI, when executing, reads data base from magnetic tape file called CORC1975FILE.

4 To begin with a message will appear on Console-Typewriter as given below

DISPLAY — ON 3, IF POPSI ON PRINTER
HALTED 00

Note.—If POPSI is to be obtained on on-line printer operator should type *ON* 3 on Console Typewriter and press the ACCEPT key. Press the ACCEPT key after typing *GO* on Console Typewriter.

5 Another message will appear at Console Typewriter:

DISPLAY LOAD CONTROL CARD
HALTED 01

Note.—Load control card. Design of the control card is illustrated in Sec 42.

6 If the switch 3 is off, then a message will appear at Console Typewriter as given below:

DISPLAY ON 2 IF PRINT NECESSARY
HALTED 02.

Note 2.—Operator may type *ON* 2 Console Typewriter if user requires the print out of every step involved in programming of generating indexes, such as construction of short display, introduction of " < " into Feature Heading Section, construction of Lead Table, Construction of Lead Term Section etc. Unless one finds an error in the program POPSI, it is not advisable to type ON 2 on console typewriter. Otherwise, type GO and press the ACCEPT key.

7 If the switch 3 is on, system reads POPSI. FILE and output is obtained on on-line printer.

8 At the end of the creation of POPSI FILE, the following message will appear on console typewriter:

DISPLAY SORT OUT TAPE
HALTED JOB OVER

Note.—In such case, the tape file is to be sorted. It is preferable to adopt the manufacturer's software for the purpose — XSDA in case of ICL 1900 series. The following information is necessary to prepare control cards for XSDA software:

Name of the file	POPSI FILE
Number of characters per record	340
Number of keys	2
Address of the first key	1st word (0th character)
Address of the second key	31st word (0th character)
Order	Ascending.

9 While generating subject index entries, it is quite likely that a message either] or) *MISSING*

*RECORD NO **** OMITTED* may appear on console typewriter. In such cases, data base has to be carefully re-edited, and the program run again so that subject index entries may be generated for all the entries.

10 A message *SHORT-DISPLAY BEYOND 210 CHARACTER* may appear at Console Typewriter. In such cases, data base has to be edited suitably such that short subject representation does not go beyond 210 characters. Then program has to be run again to generate subject index entries.

8 Implementation

The programs are written in COBOL language and were implemented on ICL 1901A computer, with memory size 16 K words with 4 bytes per word, 6 bits per byte or one character. The following peripherals were used:

1 ICL 2105 Card Reader
2 ICL 1920 Card Punch
3 ICL 2405/2 Line Printer
4 ICL 1971 Magnetic tape: at least 2 tape units are necessary
5 ICL 2821 Magnetic disc (twin exchangeable disc stores; two units)

The programs are available at DRTC.

91 Conclusion

The computer program discussed herein is the beginning of step towards computerisation of POPSI. Before one attempts further development in this line, POPSI entries are to be generated for a large size machine-readable bibliographical data base.

92 Bibliographical References

1 Sec 31 BHATTACHARYYA (G). POPSI: Fundamentals and procedures. (Paper presented at DRTC Seminar on Subject Indexing. (1975) (Bangalore)).

2 Sec 31 NEELAMEGHAN (A) and GOPINATH (M A). POPSI in practice. (Paper presented to DRTC Seminar on Subject Indexing (1975) (Bangalore)).

3 Sec 31 ——, ——. Postulate-based permuted subject-indexing. (POPSI). (Lib Sc. 12; 1975; Paper H).

4 Sec 02 RANGANATHAN (S R). Prolegomena to library classification. Ed. 3. Assist by M A Gopinath. 1967. Asia, Bombay Sec CR7.

5 Sec 01 RAVICHANDRA RAO (I K). Computer generation of alphabetical subject index headings from feature heading. (Lib Sc. 10; 1973; Paper B).

6 Sec 01 Sec 31 ——. Computer-based subject indexing. (Paper submitted to DRTC Seminar on Subject Indexing. (1975) (Bangalore).).

7 Sec 21 ——. Generalised approach to computer-based information storage, retrieval and issemination. 1975. (Project submitted in part fulfilment of the requirements for the Associateship in Documentation of the Indian Statistical Institute).

93 Sample Input

92 GV,21
MICROBIOLOGY, BACTERIA; CELL, WALL; DECOMPOSITION-CH
EMICAL>DRUG>ANTIBIOTIC

93 GV,21
MICROBIOLOGY, BACTERIA, CELL, WALL;STRUCTURE ; STUDY
-INSTRUMENT>MICROSCOPE>ELECTRON MICROSCOPE

94 GV,22
MICROBIOLOGY, BACTERIA, CELL, WALL, MEMBRANE; STRUCT
URE

95 J
HORTICULTURE,FLOWER; COLOUR; DEPTH; VARIATION; DEGRE
E; EVALUATION

96 L,32
MEDICINE, HUMAN BODY>CIRCULATORY SYSTEM>HEART; STIMU
LATION-CHEMICAL>DRUG>ANTIBIOTIC

97 L,45
MEDICINE, HUMAN BODY> RESPIRATORY SYSTEM> LUNG; DISE
ASE-BACTERIA>MICOBACTERIUM- TUBERCULOSIS; TREATMENT
- CHEMICAL> DRUG> ANTIBIOTIC

98 L,45
MEDICINE, HUMAN BODY>RESPIRATORY SYSTEM> LUNG; DISE
ASE-BACTERIA>MYCOBACTERIA>MYCOBACTERIAM TUBERCULOSI
S1 -TUBERCULOSIS ; TREATMENT-CHEMICALS>BIO SUBSTANC
ES>DRUG>ANTIBIOTICS

99 LX3,631
PHARMACOLOGY, CHEMICAL>DRUG>ANTIBIOTIC;STIMULATION-C
IRCULATORY SYSTEM>HEART; STUDY-ANIMAL>RABBIT

100 V
HISTORY, SOUTH AFRICA, HOME POLICY-(TOWARDS) INDIAN
NATIONAL
ANTI-INDIAN CAMPAIGN IN SOUTH AFRICA,1924,(23:111) (

```
15 JUNE 1976 WINTER,INDIA
  METEOROLOGY,CLIMATE;STUDY;INDIA*WINTER=1976=JUNE=15=MORNING=EARLY     88

1795 TO 1910,SOUTH AFRICA
  HISTORY,SOUTH AFRICA*1795 TO 1910    104

1860,INDIAN FRANCHISE
  HISTORY,INDIAN FRANCHISE *1860    103

1890 TO 1964,CONSTITUTION
  HISTORY,SOUTH AFRICA;CONSTITUTION*1890 TO 1964    106

1976 WINTER,INDIA
  METEOROLOGY,CLIMATE;STUDY;INDIA*WINTER=1976=JUNE=15=MORNING=EARLY     88

2 DEGREE CENT,PRESERVATION,MANGO
  FOOD TECHNOLOGY,MANGO=DISEASE;STORAGE LIFE (INFLUENCED BY)&MANGO ;PRESERVATION=COATING=ANTIMICROBIAL WAX=PACKING=POLYE
    THELENE BAG=STORAGE=2 DEGREE CENT=3 WEEKS     90

3 WEEKS,PRESERVATION,MANGO
  FOOD TECHNOLOGY;MANGO=DISEASE;STORAGE LIFE (INFLUENCED BY)&MANGO ;PRESERVATION=COATING=ANTIMICROBIAL WAX=PACKING=POLYE
    THELENE BAG=STORAGE=2 DEGREE CENT=3 WEEKS     90

ADENO CARCINOMA
  ADENO CARCINOMA;DEFINITION      7

AGE,DISTRIBUTION,INCIDENCE,CANCER
  CANCER;INCIDENCE;DISTRIBUTION=(BY) AGE     18

AKALI RELIGION,EVALUATION,NON VIOLENCE
  POLITICAL SCIENCE,NON VIOLENCE;EVALUATION=(BY) SIKH=AKALI RELIGION    105

AKALI,FREEDOM OF BELIEF AND CONSCIENCE,INDIA
  HISTORY,INDIA;FREEDOM OF BELIEF AND CONSCIENCE=SIKH=AKALI    107

ALCOHOL,CANCER
  CANCER;ALCOHOL     34

ANIMAL,STUDY,STIMULATION
  PHARMACOLOGY,ANTIBIOTIC;STIMULATION=HEART;STUDY=RABBIT     90

ANTIBIOTIC,DECOMPOSITION,WALL
  MICROBIOLOGY,BACTERIA;CELL,WALL;DECOMPOSITION=ANTIBIOTIC     92

ANTIBIOTIC,PHARMACOLOGY
  PHARMACOLOGY,ANTIBIOTIC;STIMULATION=HEART;STUDY=RABBIT     90

ANTIBIOTIC,PRODUCTION,BACTERIA
  MICROBIOLOGY,BACTERIA;PRODUCTION=ANTIBIOTIC;INCREASE=STIMULATION=ULTRAVIOLET RAYS     91

ANTIBIOTIC,STIMULATION,HEART
  MEDICINE,HEART;STIMULATION=ANTIBIOTIC     90

ANTIBIOTIC,TREATMENT,DISEASE
  MEDICINE,LUNG;DISEASE=MICOBACTERIUM=TUBERCULOSIS;TREATMENT=ANTIBIOTIC     97

ANTIBIOTICS,TECHNOLOGY
  TECHNOLOGY,ANTIBIOTICS;PRODUCTION=FERMENTATION;CONTAMINATION ;PREVENTION     90

ANTIBIOTICS,TREATMENT,DISEASE
  MEDICINE,LUNG;TUBERCULOSIS ;TREATMENT=ANTIBIOTICS     98
```

95 Appendix F: Flow Charts For CORC

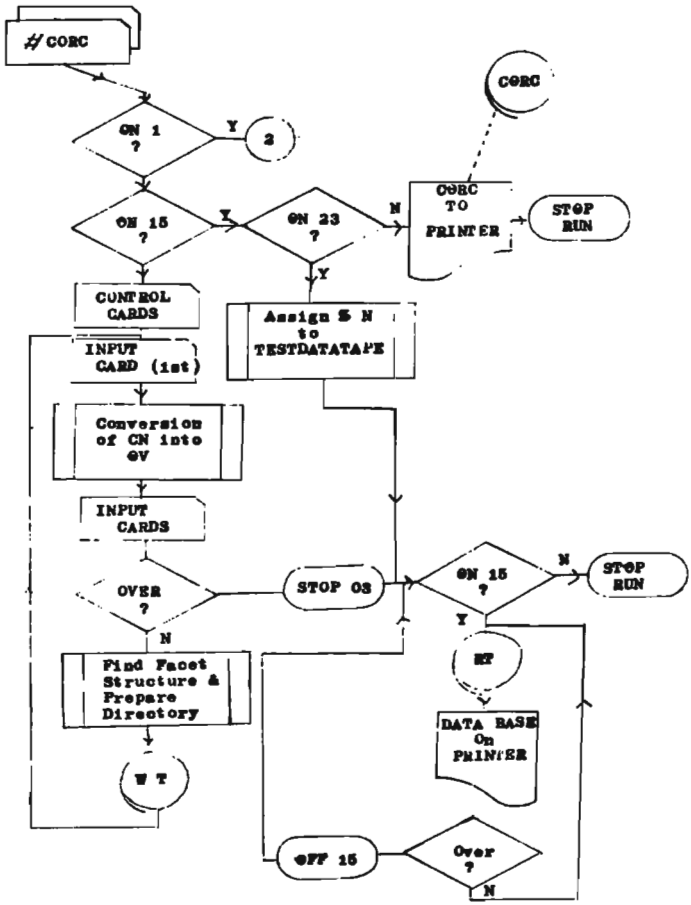

Fig. 1: Flow Diagram for CORC : to create Data Base

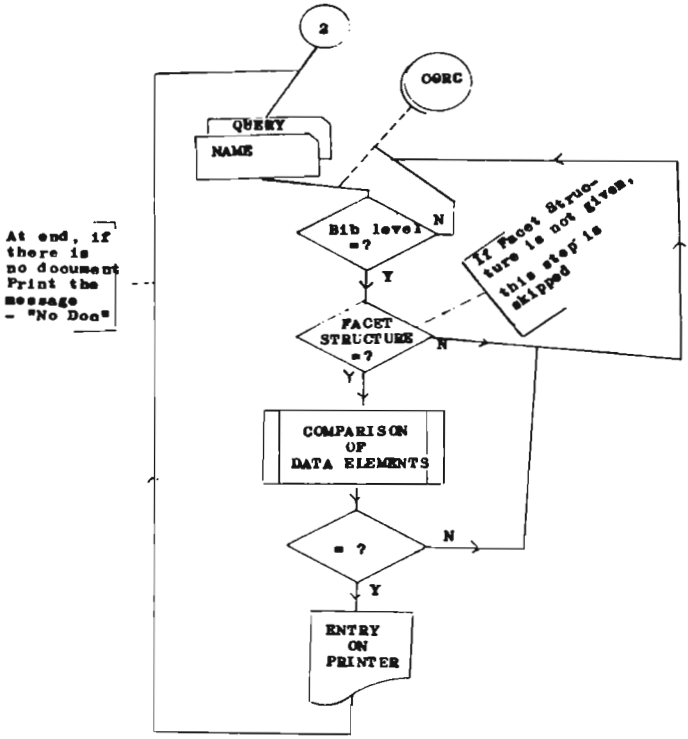

Fig. 2: Flow Diagram for CORC : to retrieve the documents

For SDBI

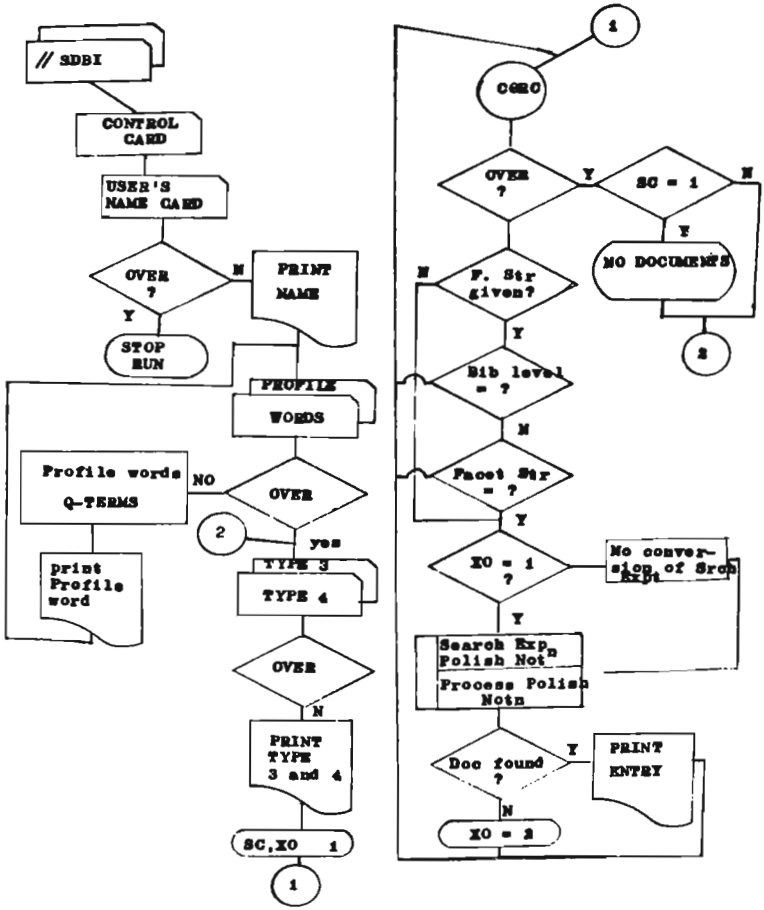

Fig. 3: Flow Diagram for SDBI