VIII. SETUP AND UPDATING OF THE CRITERION TREE FILE

Tom Evslin and Thomas Lewis

1. Introduction

The criterion tree dictionary, required as part of the syntactic processing described in Sec. VI, constitutes one file of the SMART library tape. The present section describes the construction and updating of this file consisting of relation generators, control information, and "direct" and "indirect" binary connection matrices for each criterion tree.

2. Updating of the Criterion Tree File

UPCRIT is a subroutine which creates and updates the third file of the library tape. It can be used to add items to the library, delete items, or to copy items from an old to a new library tape.

UPCRIT recognizes five control statements which may appear on tape A2. They are identified by the presence of "/SKIP", "/COPY", "/EDIT", "/ADDL", or "/WEØF" in column 1-5 of the card. The first three of the above cards may have a BCD number left justified to column 7 to indicate the number of criterion trees to be affected by the respective option. If this number does not appear, the option will be executed for all the trees remaining in the file on tape A6.

If any nonblank characters occur in columns 7-12 of an "/ADDL" card, UPCRIT will act as if it had found an EØF on A6, will refuse to look at that tape again, and will not attempt to space it past an EØF on exiting. This

option is useful when one wishes to add trees to the library from cards, and an old library tape does not exist.

"/SKIP" causes the indicated number of trees on A6 to be skipped, or, if no number appears in column 7 of the control card, the rest of the file on A6 is skipped.

"/COPY" causes the indicated number of trees to be copied from A6 to the output tape B5.

"/EDIT" causes the specified number of trees to be read from A6 and copies onto B5 after deletion of certain of the trees. UPCRIT expects from one to 30 deletion specification cards immediately following the "/EDIT" card. After each tree is read from tape A6, it is checked against all the deletion specifications. If a tree fulfills any of these deletion specifications, it is not written on B5, and an identifier for this tree is written on A3. If a tree fulfills none of the deletion specifications, it is written on B5. After the specified number of trees have been edited, a count of the number of trees actually written on B5 is written on A3.

The deletion specification cards are identified by right parentheses in columns 1 and 2. Each deletion specification card contains a serial number in columns 7-12, or a BCD index in columns 13-18, or both. A tree will fulfill a deletion specification if

(1) both the index field and the serial number match the specification; or

(2) the index (columns 13-18) is unspecified and the serial numbers match; or

(3) the serial number (columns 7-12) is unspecified, and the BCD index fields match.

Thus, for example, a tree with index ABCDEF and I.D. (serial) number 1 will match the deletion specifications

Column Number	12...7...1213...18
))	000001 ABCDEF
))	000001
))	ABCDEF

but not the specifications

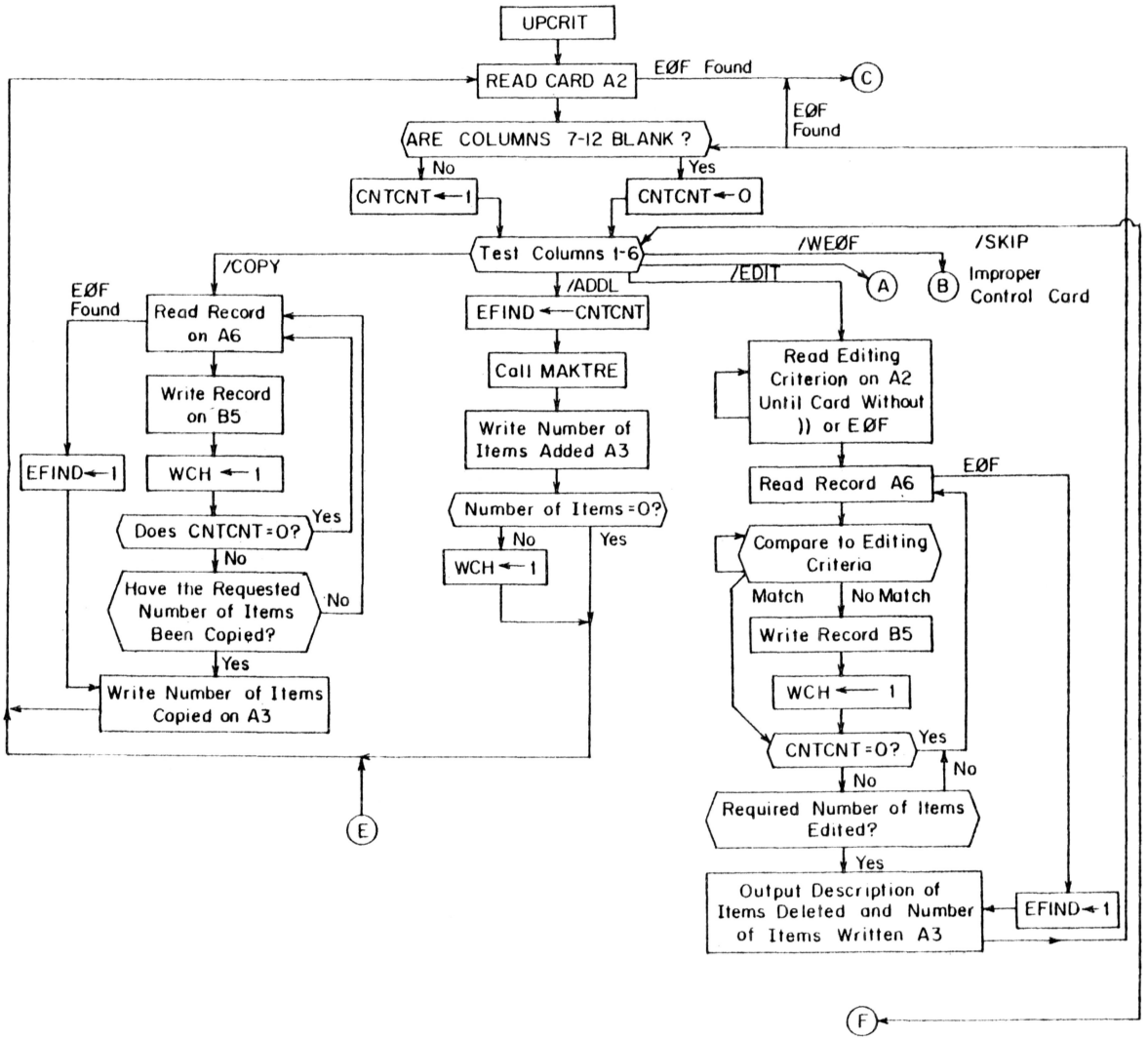
))	000001 ABCXYZ
or	
))	000002 ABCDEF.

"/ADDL" causes UPCRIT to call a routine named MAKTRE. MAKTRE reads card images from tape A2 and writes trees on B5 as specified by these card images. (A description of MAKTRE follows in Part 3 of this section.)

"/WEØF" terminates the processing. This card causes UPCRIT to write an end of file on tape B5, and positions tape A6 at the beginning of the next file. UPCRIT then exits to the main program. If no trees have yet been written on B5, UPCRIT writes a "null" record of three-zero words on B5 before the end of file.

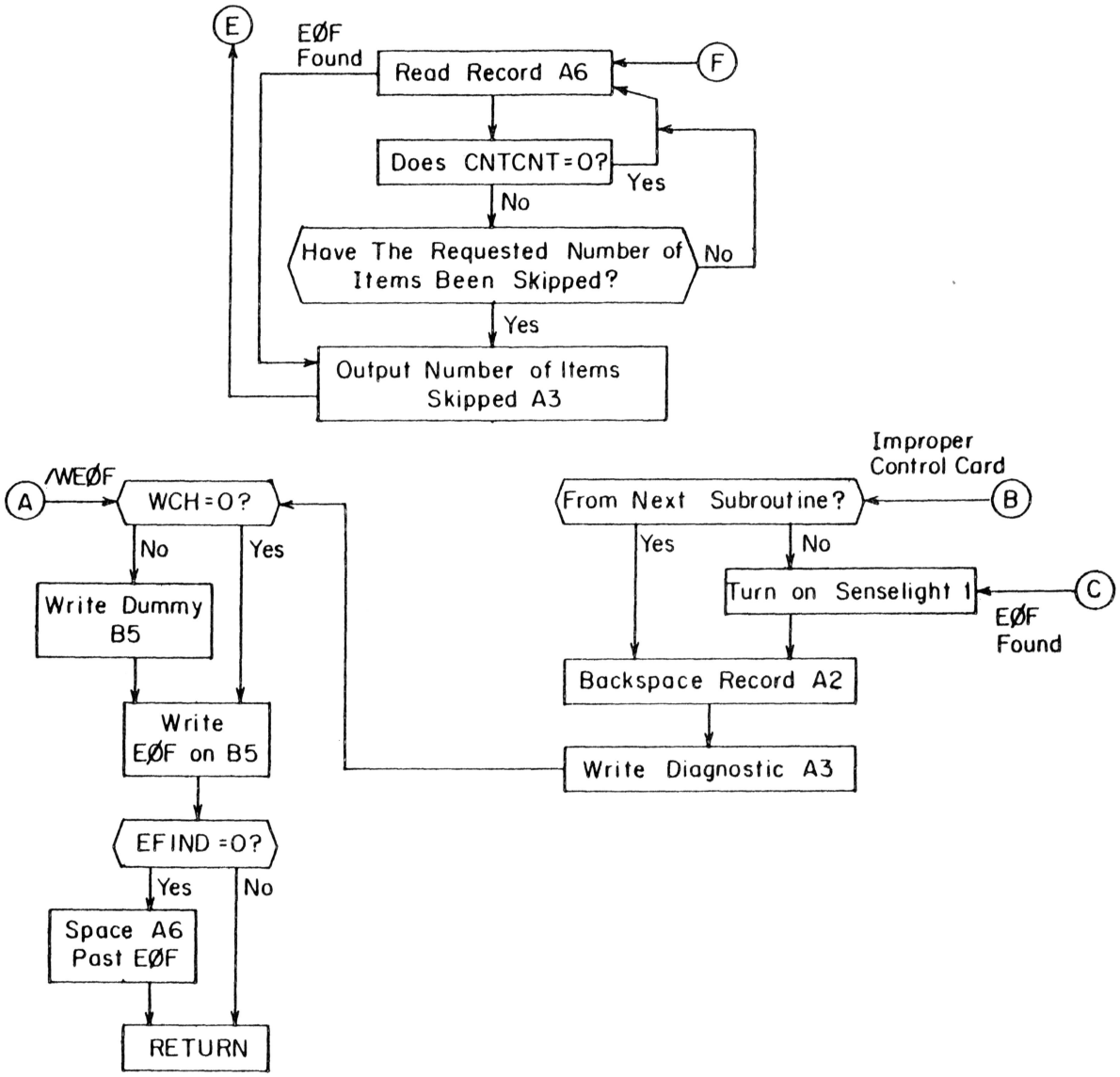
If a major error occurs, i.e., one which will interfere with later processing, UPCRIT turns on sense light 1 and writes pertinent comments on tape A3. Then UPCRIT returns immediately to the main program.

The updating procedures are illustrated by Flowchart 1.



Updating of the Criterion Tree File

Flowchart 1



Flowchart 1 (continued)

3. Construction of New Criterion Trees

MAKTRE is a FORTRAN routine which constructs criterion trees. It reads card images from tape A2 and constructs criterion trees which are written as binary records on tape B5. The cards are read in groups, each group giving rise to one criterion tree record.

This part deals only with the construction of the criterion tree file. No attempt is made here to explain the use or purpose of the various parts of a criterion tree, since these are explained in Sec. VI of this report.

A. Format of the Input Cards

Two distinct types of cards are used to specify the criterion phrases: heading cards and node cards. The heading card pertaining to a given tree contains an alphabetic code representing the subject heading to which that tree pertains. The node cards, following the heading card contain information specific to the various nodes included in a tree. A heading card followed by up to 36 node cards constitutes a card group, and any number of card groups may be used with the program during a single run. Each card group will correspond to a single binary record representing a single criterion tree on the dictionary tape.

The heading card of a card group contains a maximum of six characters located in card columns 1-6. An identification number may also appear in card columns 76-80. If such an identification number appears, the same number must appear on all following node cards, until the next heading card

is reached. Otherwise, the group will be considered incorrect and will not be written onto the dictionary tape. No heading card columns, other than those mentioned above, are processed.

Card columns 1-6 must be blank, since this distinguishes heading cards from node cards. Columns 7-8 contain the node number, right justified in this field. Node numbers must be consecutive from card to card, beginning with one for the first node card; in the case of "continuation cards," these two card columns are ignored. Nonconsecutive node numbers or node numbers greater than 36, are illegal and result in an error message. The corresponding tree is not written on tape.

Column 9 is not processed. Columns 10 and 11 contain an "antecedent" number, which may not exceed the number of node cards present. Column 12 may contain any of the four characters I,D,X,*. The symbol * denotes a continuation card which supplements the information on the preceding card. The character I causes an entry to be made in the "indirect" matrix: a "one" bit is placed at the intersection of the row indicated by the node number (columns 7-8), and the column position indicated by the antecedent number (columns 10-11). Character D causes a similar entry to be made in the "direct" matrix. "X" is a null character which does not cause any entries to be made.

Card columns 13-18 must be blank.

Columns 19-72 of the card contain the node information, and the length of this field is variable. The first nonblank character appearing in column 19 or later initiates the field. This character must be a left-parenthesis; otherwise, an error condition is recognized. The field is

terminated by a blank following a right parenthesis. Blanks occurring between the first left parenthesis and the final right parenthesis constitute an error condition and may cause premature termination of the scan.

The information appears according to the following format definitions: a pair of parentheses defines a relation. Items inside the parentheses separated by commas constitute relation generators. Thus, (A,123,BC) is a relation containing three relation generators. Strings consisting of any number of relations containing any number of relation generators may appear on a single card, provided the right parenthesis of a relation is the final character on the card. Any number of continuation cards may be used for further relations. Continuation cards have an * in column 12, relation generators in columns 19-72 (beginning with a left parenthesis), and an identification field in columns 76-80.

The following restrictions apply to the format of relation generators. Both numbers and letters are acceptable to denote relation generators; however they may not be used jointly in the same relation generator. Thus (AB,12) is valid, but (A2,B1) is not. Moreover, up to two letters or four numbers constitute a valid relation generator. Thus (AS, A,1, 12, 123, 1234) are all valid relation generators, but (ABC,12345) are invalid. A relation generator may not be zero or blank. Relation generators may consist of any of the characters in the 48 character FORTRAN set.

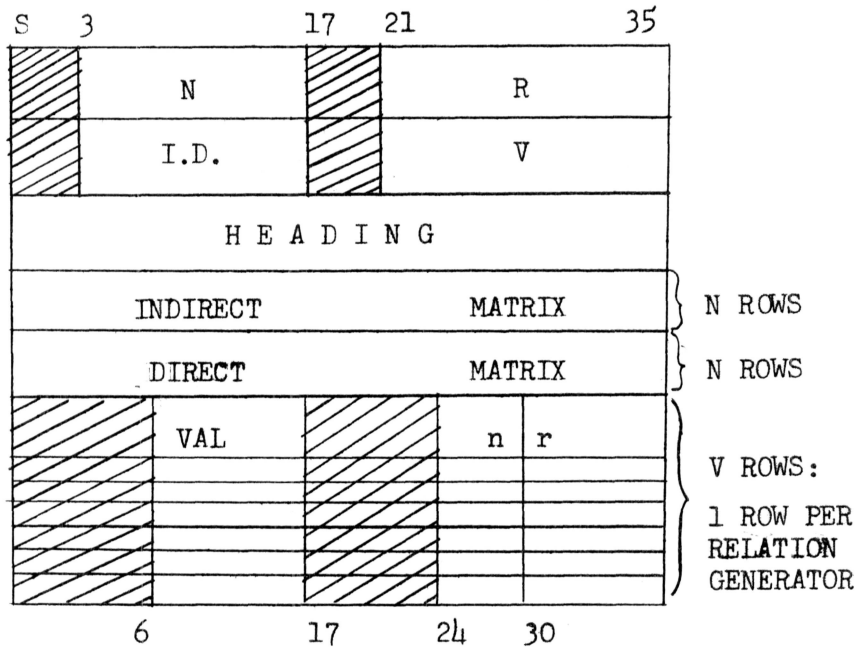
A listing of data cards, together with errors detected, is written on tape A3, the system output tape. Messages indicating the nature of the error are written out beneath the erroneous card image. Following detection of an error, further errors in the same card and the following cards of a group are

detected if possible, but the erroneous card group is not written on the dictionary tape. Card groups which pass all tests are included in the dictionary; those found unacceptable are omitted.

B. Format of the Criterion Tree File

Each card group corresponds to one binary tape record of the format shown in Fig. 1. The decrement of the first word contains the number of nodes, and the address of the first word contains the number of relations. The second word contains the identification number, if present, in the decrement (zero if none present) and the number of relation generators (see Sec. VI) in the address. The third word is the six character name taken from the heading card.

Following these three words are the "indirect" and "direct" matrices corresponding to the criterion tree, in that order. Each binary matrix contains as many words as there are nodes in its card group. Following the direct matrix are a number of words, equal to the total number of relation generators in a card group. Each relation generator corresponds to one computer word, with the following format: bits 6-17 are the 12-bit binary representation of the relation generator (preceded by a blank if the relation generator is a single character), bits 24-29 contain the number of the node to which that relation generator pertains, bits 29-35 contain the number of the relation, the number 1 being assigned to the first relation on the first card with sequencing continuing consecutively throughout the card group.



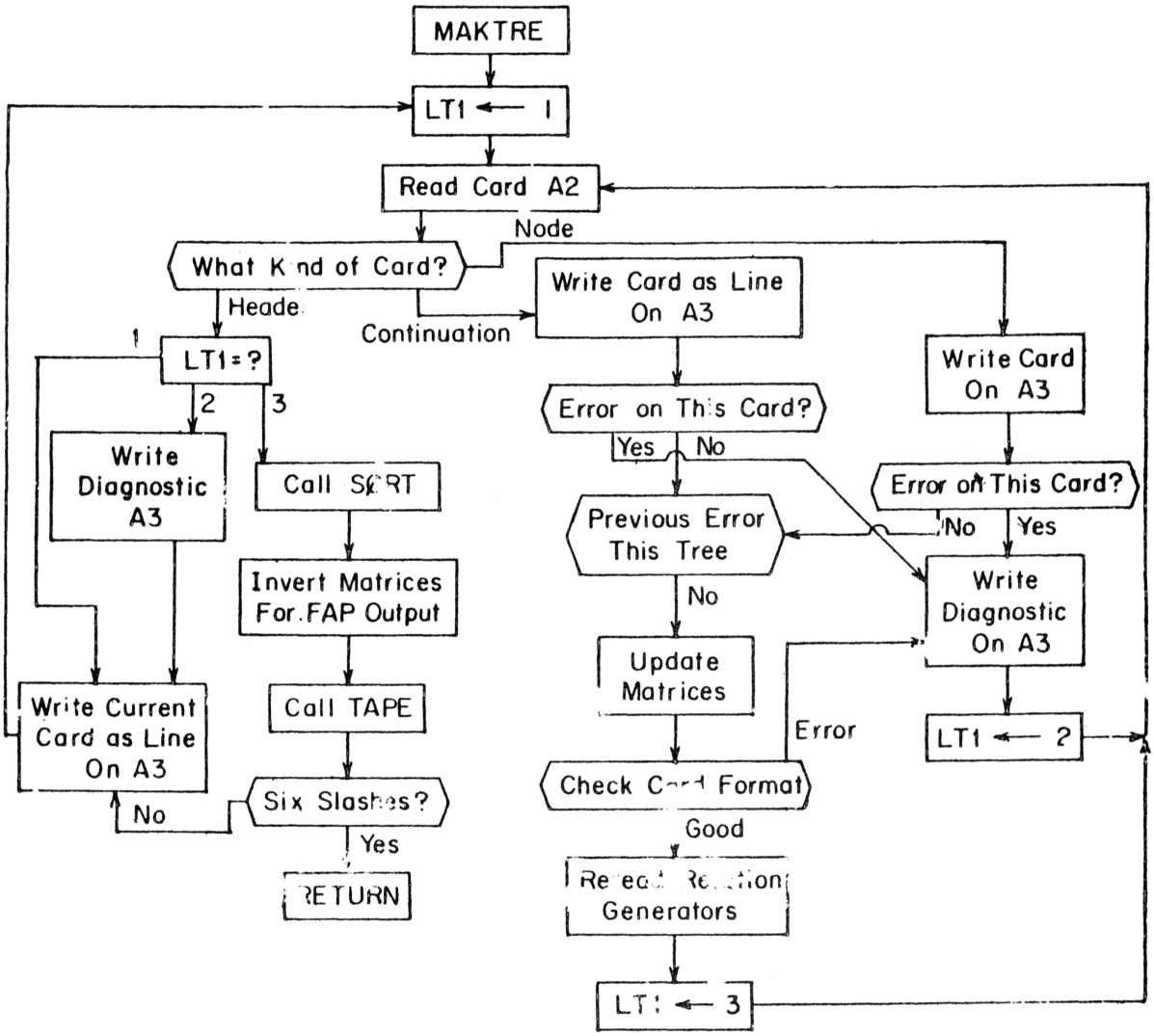
KEY:

N = Number of nodes
R = Number of relations
V = Number of relation generators
I.D. = Column 76-80 serialization
VAL = Binary value of relation generator
n = Node number of relation
r = Sequence number of a relation generator

Dictionary Tape Record Format

Figure 1

The end-of-data signal is a heading card with slashes in columns 1-6. When this card is encountered, the last card group is written on the dictionary tape. Thus, the heading word consisting of six slashes is an otherwise invalid heading symbol but must be present for proper termination of the job.



Construction of New Criterion Trees

Flowchart 2

The dictionary tape preparation program returns the number of trees written on the binary dictionary tape in the decrement portion of the accumulator. No end of file is written on the dictionary tape, nor is it rewound either at start or finish by subroutine MAKTRE. Flowchart 2 describes the MAKTRE routine.