(18) Newman, S. M., "Storage and Retrieval of Contents of Technical Literature, Nonchemical Information," p. 6, Second Supplementary Report, Patent Office Research and Development Reports No. 12, United States Department of Commerce, 1958.
(19) Richards, I. A., and C. Gibson, "Interpretation," in I. J. Lee, Ed., *op. cit.*, p. 162.
(20) Cohen, S. M., C. M. Lauer, and B. C. Schwartz, "An Evaluation of Links and Roles as Retrieval Tools," J. CHEM. DOC. 5, 118 (1965).
(21) Riddles, A. J., "Computer Based Concept Searching of United States Patent Claims," IBM Technical Report, ITIRC-004, Yorktown Heights, N. Y.: Thomas J. Watson Research Center, 1965.

Computer Assisted Primary Index Preparation

C. J. MALONEY, S. BRYAN, and M. EPSTEIN
National Institutes of Health, Bethesda, Md. 20014

Received June 28, 1966, and September 21, 1967 (Revised)

A man-machine primary indexing system employing a special "indexing language" called BICEPT is fully described. The system is applicable for computer-processed or non-machine-readable text. It differs from most other machine indexing projects in that the specification of index terms is largely manual. The method was applied to a journal article in an operational test.

PRIMARY *vs.* SECONDARY INDEXING

The use of various forms of indexes to find or to relocate desired information is of long standing and wide application. It is not generally appreciated, however, that indexes are employed in two quite distinct circumstances which we are here distinguishing by the terms "primary" and "secondary." A primary index is an index to a single document or series of documents issued from a single source (or cooperating group of sources) and normally bound with or at least an item in the collection indexed. It will, of course, be indexed at the source, and there will be no acquisition or language problem, even to the extent of technical jargon or drift in meaning of terms over time. The index in the back of any book, or the index volume of a set of encyclopedias would be examples. By secondary index we mean an index normally providing access to a multiplicity of documents issued from a variety of sources and over an extended period of time, usually planned to extend into the future. The index to *Chemical Abstracts* and the *Grants Index* of the National Institutes of Health are two examples. An annual or decennial index to a newspaper or journal could fall in either class, but is usually intermediate between them.

A secondary index answers the question: "which (if any) documents discuss the subject of....?" A primary index answers the question: "where (if at all) in *this* document is a statement, however brief, disguised, and/or trivial, made on the subject of....?"

A number of manuals and one journal (*1*) are specifically devoted to preparation of primary indexes, though these are more likely to stress similarities than differences between the two classes of indexes. A fairly full comparison of these two related but distinct problems is given elsewhere (*2*). A most important distinction, however, which makes discussion of computer assisted preparation of primary indexes especially timely is that, increasingly, the full text of the item to be indexed is available in machine-readable form. For example, the American Chemical Society plans to produce all of its journals by computer by the early seventies (*3*). The full impact of this now well-established trend is being appreciated only slowly. Thus Baxendale (*4*) in her otherwise excellent review of the 1965 status of "content analysis, specification, and control" includes "the current cost of providing computer-readable input" in the cost of primary indexing.

Another distinction relates to the fullness of indexing in the two situations. Judgments concerning the desirability of full indexing for one purpose cannot be successfully based on considerations arising in the uses for the other, yet until the two applications are distinguished there is a not unnatural tendency to do so. Judged by standards applicable to a secondary index, a primary index will appear unduly prolix. Its merits can only be appreciated by one who has spent hours in attempting to relocate a buried item in a book, journal, or newspaper, which he is sure he read but wishes to relocate. *The authors feel it is quite safe to assert that, when very full indexing is technically feasible to supply, the demand for doing so will be found to exist, as was true when computers made extensive computation possible, or when the aeroplane became efficient for long distance travel.* Indeed, certain authors (*5*, *6*) have considered it worthwhile to study the profitability of storing full text and then scanning the total store in answer to every query. Such an approach may be justified in case the total potential of the file must be exhausted upon enquiry—say, in a tactical military situation. It is quite possible that, in certain other

situations, speedier response or some degree of storage compression would suggest an indexing approach but at a much fuller level than would be suitable for most applications. Doubters of the value of detailed indexing have not been lacking. Both extremes cannot be right. There is room for an intermediate solution such as provided here.

A third major concern is the role of the computer in the index formulation process. Full text retrieval systems dispense with the index altogether. So while using the machine for retrieval, they do not use it to form an index. The most detailed "indexing" is concordance formation. Since 1941 only one hand-produced concordance has been published; all others were computer produced (*7*). [The authors found this statement in Reference (*9*). The reader is invited to try to do so using the 18-page subject index of the book.] Index sorting alone has justified the use of the computer in certain cases (*8*). It is our view that any computer type-set book which is to be provided with a detailed index is a fitting candidate for the technique.

OUTLINE OF APPROACH

In the preparation of either form of index (primary or secondary), the actual process consists of an interplay of decisional and of operational steps. The decision making is an intellectual process and, hence, is ill adapted to machine execution both from the subject matter and from the linguistic analysis standpoints. The operational steps are manipulative functions and, hence, well adapted to computer execution. The technique described in this paper seeks to employ human judgment, (a) for completing the rejection of nonindexable material, (b) in delimiting phrase boundaries for subunits of index entries, (c) in indicating which phrases are to compose the ultimate index entries, (d) in editing the index entries, and (e) in supplying supplementary entries. It is contemplated that, initially, screening standards would be set low, with a final rejection and consolidation step being postponed until the alphabetized index begins to take shape. By this last procedure, many of the most difficult decisions the indexer must make will be postponed until he is in the best position to make them. This is believed to be a major advantage over purely manual indexing. The added computer processing is negligible. Step (a) is performed in part by computer in the current version, but the future should bring a succession of extensions. Step (b) above may, in the reasonably near future, be transferred to the computer, subject to human surveillance. Steps (c), (d), and (e) likewise may eventually be tentatively performed by the computer subject to manual overriding, as soon as the labor so required is less than now involved in performing the positive operation. This would constitute a form of "indexing by exception" with all the benefits implied, yet avoid a total reliance on the computer.

The procedure described in the sequel has been implemented only in part, as this is an interim report. Operations involving no programming complexities will be presumed, though; in a later section, the exact current status will be detailed. Partial application to a specific trial is described below.

It is expected that the text to be indexed will have already been computer processed, either as an editing step (including revision or updating), a typesetting step, or both. Indexing will be made from a computer listing, hopefully produced on the same computer run as one of the above operations. Each sentence is automatically numbered sequentially by the computer throughout the text, or a major subdivision of the text, such as a chapter. Each word within a sentence similarly is supplied with a word number which, however, is omitted in the output if the word is recognized by the computer as not being a possible index entry. These nonentry words in the form of a dictionary are read into memory prior to the tagging run (*10*).

The manual steps in the initial processing of the first printout fall into three distinct types, not necessarily performed by three different people. The first is intellectually the most difficult, but operationally the easiest. It consists in marking the printer output copy to show (1) deletions, (2) additions, (3) entry points, and (4) index modifiers as described in full detail in the next section. Only the most trivial special conventions must be learned to perform this step as outlined in our example. In fact, any modification of the conventions given in this paper which seem appropriate to the indexer can be adopted, including standard proofreader's marks where applicable, as the markings are converted in the next (coding) operation prior to being introduced into the computer.

It is, of course, possible to obtain a complete printout in text order with the successive chosen index entries listed following each sentence for a check of completeness and accuracy before calling for the sorting run, which, while listing the index entries in alphabetical order, is (presumably) still not supplied with page (and line) reference, but only sentence and word. This listing, however, is very valuable to the indexer in reworking, consolidating, and adding entries.

After page make-up, the sentence and word number beginning (ending) each page is entered into the computer, which then performs an automatic decoding for the printed index. Index production need not be delayed for pagination, but this step can overlap author's correction of galley proofs. In the final printed index, location of entries can be specified down to page, line, and word within line, at the cost of maximal index bulk, or any chosen level of reference can be chosen. *Line* numbers would *not* have to be supplied to the computer—only page numbers—whether the index entries as finally published supply line numbers, section of page, or neither. In the system where fifths of a page are denoted by the letters *a* through *e*, the computer can be set to make this conversion, or alternatively compression could be specified to occur only where the same word occurs twice in the same sentence (line).

The completed index would be computer type-set. We have not attempted this step in our studies but it appears straightforward. It is mentioned here because automatic input and output to computer typesetting of the index are important pluses to the use of the computer in generating the index.

THE BICEPT LANGUAGE

This section describes a special "indexing language" called BICEPT formed as an acronym from *book indexing with context and entry points from text.* BICEPT consists of computer interpretable statements which may be formed efficiently to obtain an index by composing index entries of edited text phrases. Formation of BICEPT is a straightforward clerical operation which, while taking a short time to learn, proceeds rapidly in execution. BICEPT is a man-machine system. The manual steps are:

Marking a computer-produced listing of the text.
Describing these markings in BICEPT language.
Keyboarding each BICEPT statement.

The machine steps are:

Producing the computer listing in the first step above.
Interpretation of BICEPT statements as fetching and editing operations on text phrases.
Supplying page (and subpage) reference numbers.
Alphabetically sorting the above results.

This project differs from most other machine indexing projects in that the specification of index terms is largely manual. The machine's assistance in the BICEPT system is:

To reduce the writing and keyboarding involved in index preparation by:
a) Specifying an index entry with word-numbers and a sentence-number instead of writing it out.
b) Creating multiple index entries from a single text string by specifying multiple "entry points" instead of respecifying the same text phrase.

To supply page references, and, if desired, location on page by any chosen system of designation.
To sort alphabetically the index entries.
To serve as input for typesetting index.

ECONOMICS

Individual journal articles, or even separate issues, are not ordinarily supplied with indexes and even the provision of annual subject indexes seems not to be common. Solution of the problem of providing expeditious, economical, effective subject indexes by computer or otherwise might serve to reverse this trend with benefits which could be appreciated only when the product became available. Purely as an example illustrating the BICEPT technique, the article of A. Bondi, "On Error Prevention," which appeared in the JOURNAL OF CHEMICAL DOCUMENTATION for August 1966, pages 137 to 142, has been indexed, both manually and by computer, in somewhat greater detail than might ever prove desirable in practice.

Meaningful cost comparison requires procedures fully worked out and tested—and that is just what is not available in first attempts for any operation. Comparison can only be of the roughest sort at this time for many reasons. The input tape was the output of the IBM 1401 computer, but with a number of special coding conventions which made transference to the National Institutes of Health Honeywell 800 (and later to its IBM 360-50) difficult. The efficient common word dictionary technique previously described (*10*) was not used, also for reasons of programming economy. Finally, code conversion from sentence and word to page and line was done manually for the same reason. The results of this trial have encouraged the authors to explore the possibility of semi-operational application of the method.

Automatic input for index preparation and automatic output for computer typesetting of the index are important considerations. The improved work flow of in-house professional index preparation is also an important, if intangible, consideration.

EXAMPLE

Treatment of the first 11 words of the Bondi article (*11*) is set out below in ordered steps to make the procedure clear.

Step 1 (Computer). Create a "tagged" text listing.

1. THE ANALYSIS OF ERROR INCIDENCE AND
   2 4 5
   CONTROL IN THE CHEMICAL LITERATURE
   7 10 11

The reference number (1) to the left of the sentence and the word number printed under each word constitute computer-produced tags. Only words which might constitute entries in the alphabetized index (either directly or as synonyms), are tagged. Words which are virtually certain not to *initiate* index entries, though they may well appear within them, are *not* tagged. These noncandidate entry words are contained in a dictionary in memory. Accordingly, the indexer's attention is directed to tagged words only. As future research transfers a greater and greater portion of the work to the computer, the fraction of tagged words to total text—and hence the indexer's work—will be reduced. Should the indexer choose to select any such untagged word however, he merely treats it as he does any tagged word, and all following steps occur without further human attention.

Step 2 (Indexer). Mark listing.

a. *Delimit index string* by enclosing in square brackets an over-all section within one sentence from which one or more index entries will be formed.

   1. THE ANALYSIS OF [ERROR INCIDENCE AND
      2 4 5
      CONTROL IN THE CHEMICAL LITERATURE]
      7 10 11

b. *Edit the index string.*

1. *Delete words* by striking through or crossing out unwanted words. For example, the word "the" preceding the word "chemical" is marked for omission.

   1. THE ANALYSIS OF [ERROR INCIDENCE AND
      2 4 5
      CONTROL IN ~~THE~~ CHEMICAL LITERATURE]
      7 10 11

2. *Insert words.* To insert the word "journal" between the words "chemical" and "literature" use the usual proofreader's correction procedure of inserting a caret in the

text and placing the added word (or phrase) in the adjacent margin (or use any convenient technique that will be clear to the coder).

1. THE ANALYSIS OF [ERROR INCIDENCE AND
2 4 5
(Journal CONTROL IN ~~THE~~ CHEMICAL LITERATURE]
7 10 ^ 11

c. *Delimit entry points.* Draw a vinculum over, an underscore under, or enclose in parentheses or braces each word (phrase) which is to form an index entry point.

1. THE ANALYSIS OF [ERROR INCIDENCE AND
2 4 5
(Journal CONTROL IN ~~THE~~ {CHEMICAL LITERATURE}]
7 10 ^ 11

The markings on the phrase "chemical journal literature" mean that the text phrase is to appear under each of these three words in the alphabetized index. Under "chemical" all three words will appear, under "journal" the last two, and under "literature" only that word itself.

Alternatively, as each of the three words "chemical," "journal," "literature" is to start an entry and the following, but not preceding, words are to be included in the entry head, the markings could be

(CHEMICAL (JOURNAL (LITERATURE)))

or any other combination of enclosure markings that would be clear to the coder. The only problem (other than the intellectual decisional problem) is to delineate nested entry words or phrases.

It is true that these concepts are rife in the publication and the first sentence says little that is unique. It is likewise true that, when this is so, only a line number (and possibly a page number) at most need be added to the bulk of the index. As the index is to be *complete*, scanning neither of the index nor of the text is to be required even when minutia are sought. This is the primary difference in the roles of the two forms of index.

Step 3 (Coder). Describe the text markings in BICEPT language.

The complete description of BICEPT is given in a later section. The BICEPT statement for the marks shown after step 2c is

1. (4, 7) 8-(10 (JOURNAL (11)))

In effect, the word number (or the word itself) is written or implied in the BICEPT statement for every word of the sentence which is both between the brackets and to appear in the index. Words which are between the brackets but are *not* to appear are replaced by dashes. Words (phrases) which are to serve as entry points are enlcosed by nested parentheses.

Step 4 (Keyboard Operator). Keyboard the BICEPT statement.

A minimum of special instructions are required for keyboarding. At the cost of using special coding sheets, fully filled in, special instructions can be rendered entirely superfluous.

Step 5 (Computer). Execute the BICEPT statement.

a. Interpret the statement.

The program fetches words 4 through 11 of sentence 1 from the text file (same as used in step 1). Next this phrase is edited. The dash applies to the one word "the" between index words 8 and 10, and indicates that it is to be omitted from every index entry formed from this sentence. The string "JOURNAL" is detected as being neither a number nor punctuation and is thus interpreted as an inserted word after word 10 (the number on its immediate left). From the original text the program thus assembles a string in memory as

1. ERROR INCIDENCE AND CONTROL IN CHEMICAL JOURNAL LITERATURE

b. Extract phrases for entry points.

The program extracts phrases by a parenthesis matching routine yielding in this example four phrases composed of the words:

(1) 4, 5, 6, and 7.
(2) 11.
(3) JOURNAL and 11.
(4) 10, JOURNAL, and 11.

The effect of step 5 is to create the following index entries:

(1) ERROR INCIDENCE AND CONTROL*—IN CHEMICAL JOURNAL LITERATURE.
(2) LITERATURE* ERROR INCIDENCE AND CONTROL IN CHEMICAL JOURNAL—.
(3) JOURNAL LITERATURE* ERROR INCIDENCE AND CONTROL IN CHEMICAL—.
(4) CHEMICAL JOURNAL LITERATURE* ERROR INCIDENCE AND CONTROL IN—.

The asterisk marks the end of the index heading. The dashes give its location in the full text string. The indexer is thus provided all information usable in deciding whether to retain or to delete the entry in his review and revision of this preliminary output. This output may be edited manually to produce the following:

(1) Chemical journal literature error incidence and control.
(2) Error incidence and control in chemical journal literature.
(3) Journal literature error incidence and control.
(4) Literature error incidence and control.

This one context will thus be found whether the searcher elects to look under chemical, error, journal, or literature. Lest it be decided that it is not worth the cost to afford him such elaborate aid, the "cost" is only in the printing of the full index, *not* appreciably in its formation. This could well lead to very full indexing in manuscript indexes in newspaper, journal, or book publishing offices, with extracted entries only appearing in the printed version. Of course, unless these entries applied nowhere else in the entire item, their inclusion adds only a page (and subpage) citation.

Steps 2 through 5, described in detail above by way of example, are repeated until the complete text is processed.

Output from step 5b is recorded onto magnetic tape, which is passed to a conventional sorting routine for

alphabetization. Rules for computer alphabetization have appeared recently (*12*). The complete process is illustrated further in Figures 2 through 5 by an entire page of text.

COMPLETE BICEPT PROCEDURE

The procedure for compiling a primary index by BICEPT begins with the computer-produced listing containing numbered sentences in their entirety interlineated with word numbers of candidate index entries. The division of processing into decisional and coding phases is designed to place a minimum of constraints on the intellectual phase of specifying index entries. It is anticipated that entry choices will be made by a professional indexer, who in general will have minimal computer contact, but will be familiar with proofreader's conventions and with marked text. The procedures described in relation to the example of Section V purposely were kept simple for ease of comprehension. Extensions to the current version of BICEPT will be discussed later.

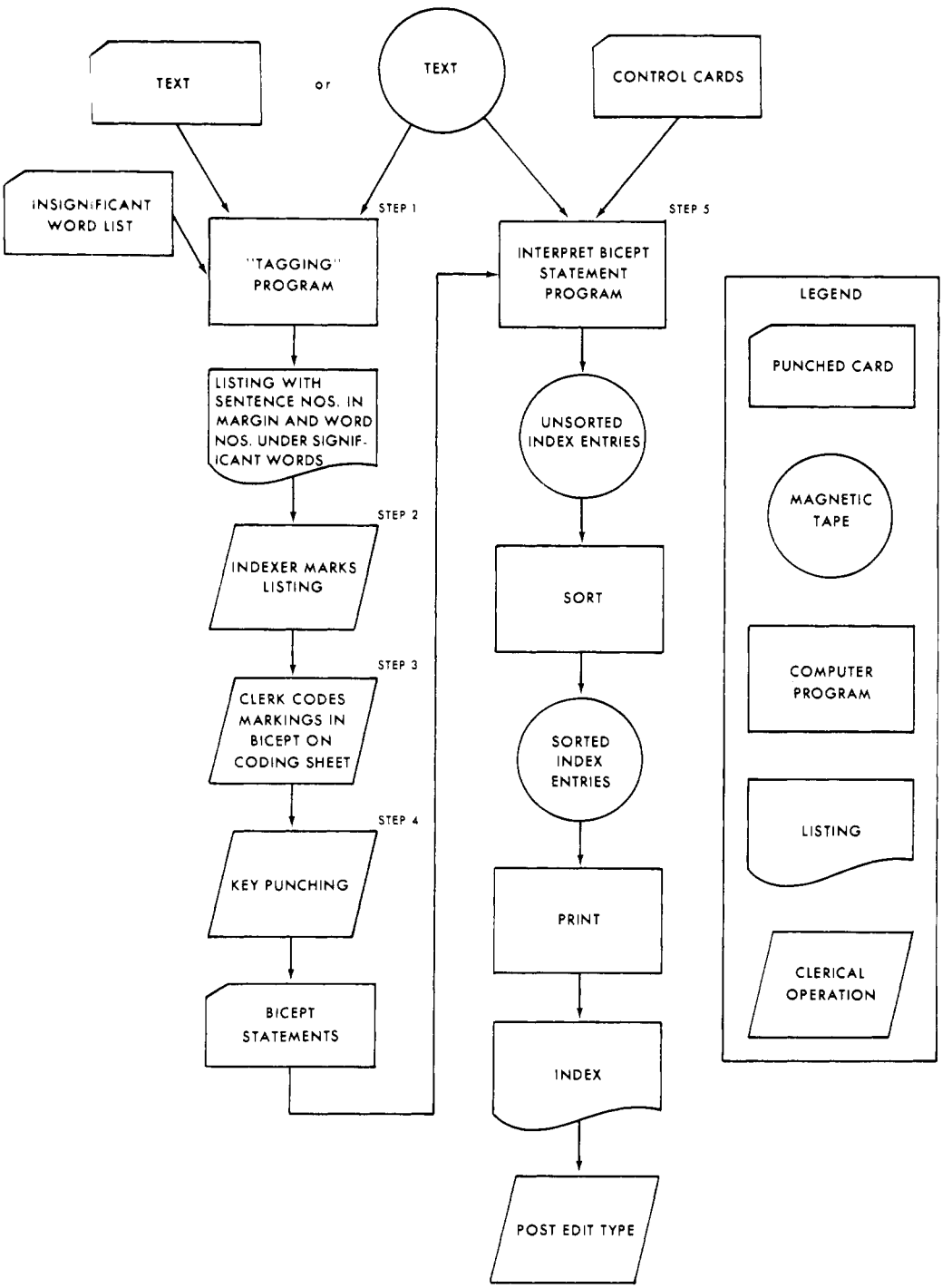

Figure 1. Process chart for BICEPT System

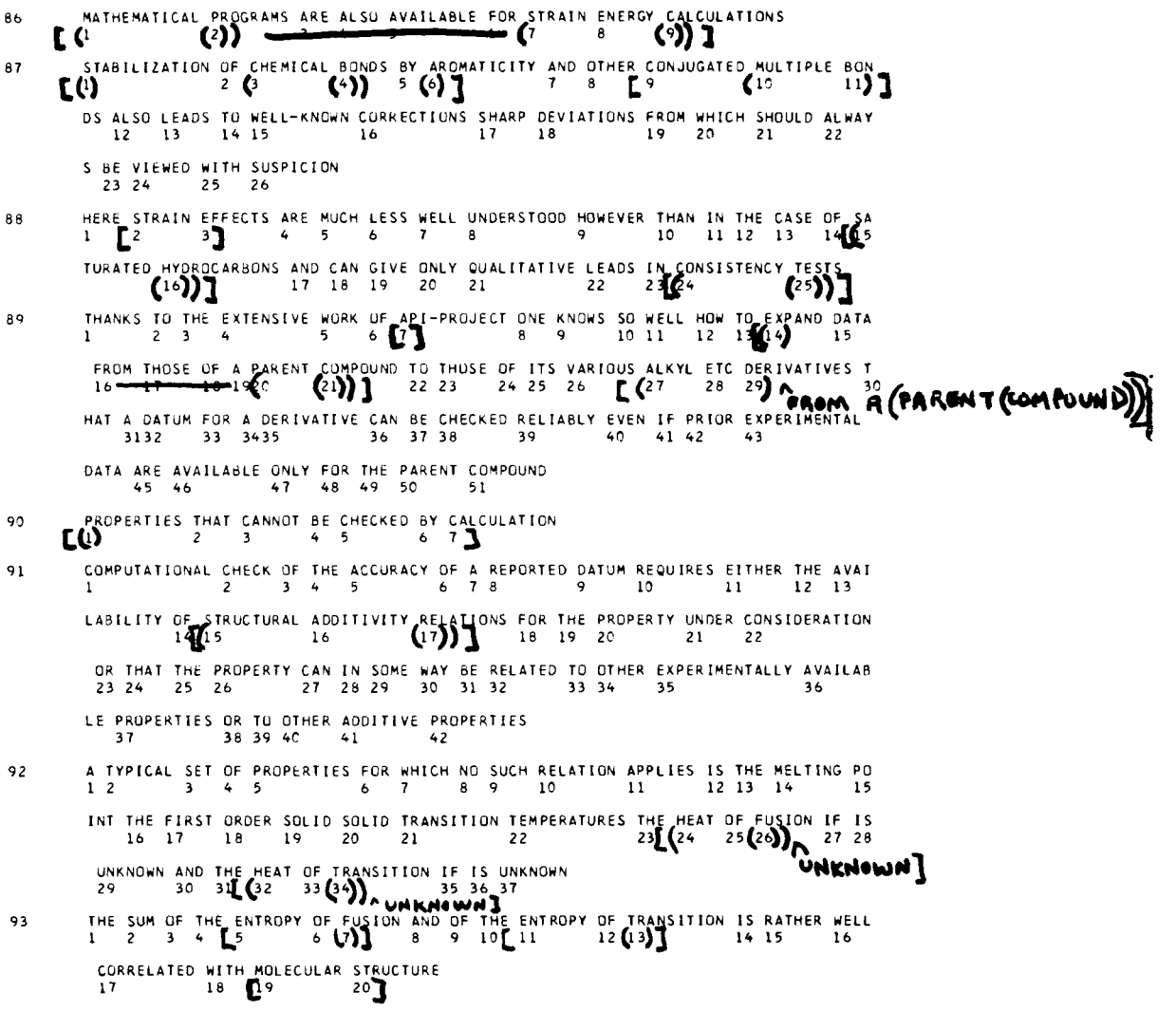

Figure 2. Page of text marked by indexer (step 2)

At present the words which will appear in the final index as printed must all occur in that form within the confines of a single sentence, or must be written (and keyboarded) even if present in the surrounding sentences. Further, the BICEPT approach delimits that section of the sentence from which a given cluster of index entries are to be formed. The choice of bracket pairs for this purpose is, of course, arbitrary. The brackets do not enter the computer. In general, the index cluster consists of a permutation of those words of any one entry that are to constitute distinct entry points.

No complication arises from two or more bracket pairs occurring within one sentence, unless these pairs overlap. Any notation which would enable the coder to recognize the indexer's intention in this situation will serve. For example, the indexer could write a small letter *a* just above the beginning and ending bracket composing the pair with the leftmost left bracket, a small letter *b* for the next pair, and so forth. To the extent required for clarity, the same superscript would be written over all markings relating to the entire cluster formed from each bracket pair.

Grouping symbols (most often parentheses) cannot extend into text enclosed by two or more brackets unless enclosed by another appropriate pair, since the function of brackets is precisely to delineate a text string from which an index cluster is to be formed. A similar system of superscripts can be used to distinguish index markings appropriate to one index entry from those appropriate to another where bracket pairs overlap, or where different words are to form two index clusters from one bracket pair. Superscript numerals may be used for this purpose.

If the markings threaten to become confusing, the indexer need merely write out any number of entries in full, attaching the appropriate sentence number.

The text lines of the marked listing may now consist of the following strings of symbols between left and right brackets:

Undeleted original text words
Deleted original text words
Inserted words
Grouping symbols
(i.e., whatever markings the indexer opted to use to denote left and right boundaries of entry points)

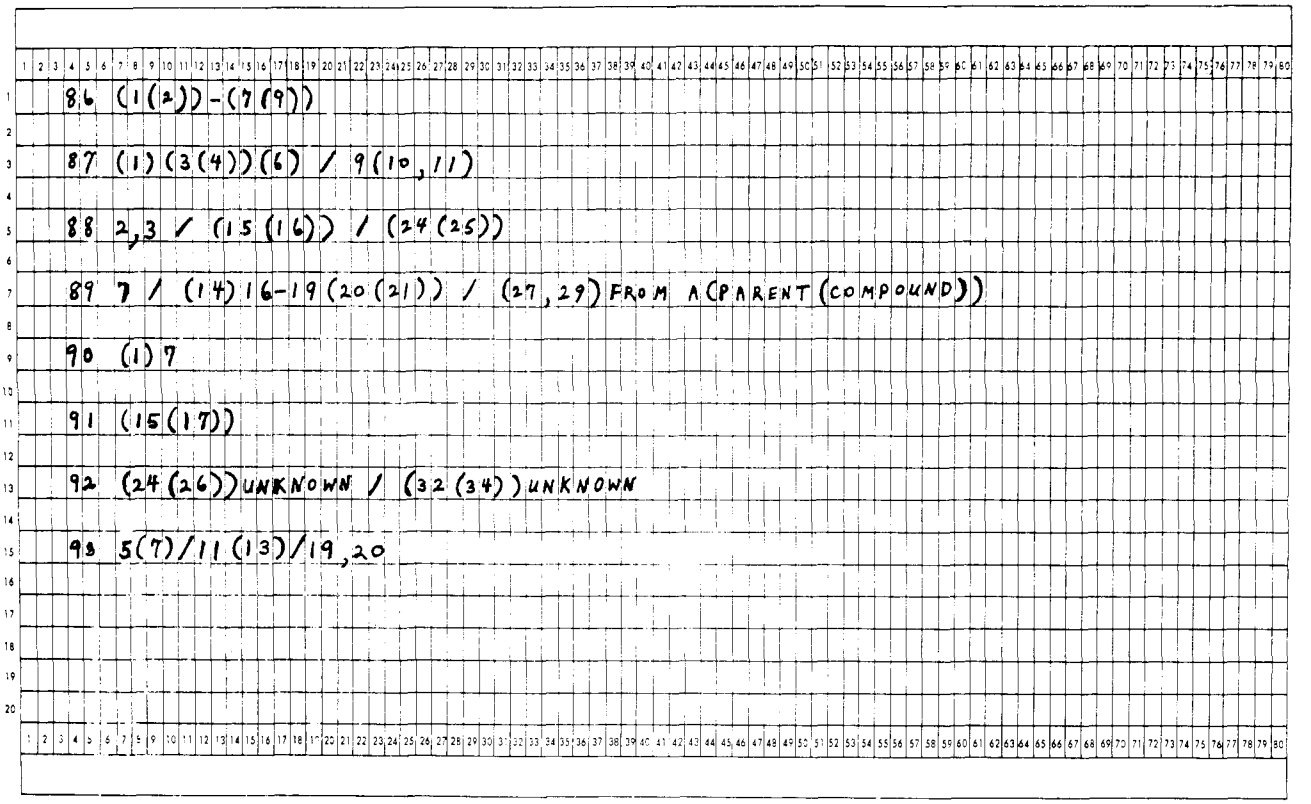

Figure 3. Markings coded in BICEPT (step 3)

```
                                BELOW IS A LISTING OF THE UNSORTED INDEX CANDIDATES
                                DEMONSTRATION RUN ON SINGLE PAGE
54   86          4     PROGRAMS *MATHEMATICAL - -  STRAIN ENERGY CALCULATIONS
54   86          4     MATHEMATICAL PROGRAMS *- -  STRAIN ENERGY CALCULATIONS
54   86          4     CALCULATIONS *MATHEMATICAL PROGRAMS STRAIN ENERGY - -
54   86          4     STRAIN ENERGY CALCULATIONS *MATHEMATICAL PROGRAMS - -
52   87          4     STABILIZATION *- -  OF CHEMICAL BONDS BY AROMATICITY
52   87          4     BONDS *STABILIZATION OF CHEMICAL - -  BY AROMATICITY
52   87          4     CHEMICAL BONDS *STABILIZATION OF - -  BY AROMATICITY
52   87          4     AROMATICITY *STABILIZATION OF CHEMICAL BONDS BY - -
38   88          4     MULTIPLE BONDS *CONJUGATED - -
38   88          4     STRAIN EFFECTS *
38   88          4     HYDROCARBONS *SATURATED - -
38   88          4     SATURATED HYDROCARBONS *
38   89          4     TESTS *CONSISTENCY - -
38   89          4     CONSISTENCY TESTS *
38   89          4     API-PROJECT *
40   89          4     EXPAND *- -  DATA FROM A PARENT COMPOUND
40   89          4     COMPOUND *EXPAND DATA FROM A PARENT - -
40   89          4     PARENT COMPOUND *EXPAND DATA FROM A - -
50   89          4     ALKYL ETC DERIVATIVES *- -  FROM A PARENT COMPOUND
50   89          4     COMPOUND *ALKYL ETC DERIVATIVES FROM A PARENT - -
50   89          4     PARENT COMPOUND *ALKYL ETC DERIVATIVES FROM A - -
54   90          4     PROPERTIES *- -  THAT CANNOT BE CHECKED BY CALCULATION
37   91          4     RELATIONS *STRUCTURAL ADDITIVITY - -
38   91          4     STRUCTURAL ADDITIVITY RELATIONS *
38   92          4     FUSION *HEAT OF - -  UNKNOWN
38   92          4     HEAT OF FUSION *- -  UNKNOWN
38   92          4     TRANSITION *HEAT OF - -  UNKNOWN
38   92          4     HEAT OF TRANSITION *- -  UNKNOWN
38   93          4     FUSION *ENTROPY OF - -
38   93          4     TRANSITION *ENTROPY OF - -
38   93          4     MOLECULAR STRUCTURE *
END OF DATA FOR SDS 0X MEMBER
```

Figure 4. Initial candidate index output (unsorted) (step 5b)

```
SEST NO  PG NO    INDEX ENTRY WITH FORMAT, ENTRY POINT(S)(CONTEXT SUBSET)...

89       4     ALKYL ETC DERIVATIVES *- -  FROM A PARENT COMPOUND
89       4     API-PROJECT *
87       4     AROMATICITY *STABILIZATION OF CHEMICAL BONDS BY - -
87       4     BONDS *STABILIZATION OF CHEMICAL - -  BY AROMATICITY
86       4     CALCULATIONS *MATHEMATICAL PROGRAMS STRAIN ENERGY - -
87       4     CHEMICAL BONDS *STABILIZATION OF - -  BY AROMATICITY
89       4     COMPOUND *ALKYL ETC DERIVATIVES FROM A PARENT - -
89       4     COMPOUND *EXPAND DATA FROM A PARENT - -
88       4     CONSISTENCY TESTS *
89       4     EXPAND *- -  DATA FROM A PARENT COMPOUND
93       4     FUSION *ENTROPY OF - -
92       4     FUSION *HEAT OF - -  UNKNOWN
92       4     HEAT OF FUSION *- -  UNKNOWN
92       4     HEAT OF TRANSITION *- -  UNKNOWN
86       4     HYDROCARBONS *SATURATED - -
86       4     MATHEMATICAL PROGRAMS *- -  STRAIN ENERGY CALCULATIONS
93       4     MOLECULAR STRUCTURE *
87       4     MULTIPLE BONDS *CONJUGATED - -
89       4     PARENT COMPOUND *ALKYL ETC DERIVATIVES FROM A - -
89       4     PARENT COMPOUND *EXPAND DATA FROM A - -
86       4     PROGRAMS *MATHEMATICAL - -  STRAIN ENERGY CALCULATIONS
90       4     PROPERTIES *- -  THAT CANNOT BE CHECKED BY CALCULATION
91       4     RELATIONS *STRUCTURAL ADDITIVITY - -
88       4     SATURATED HYDROCARBONS *
87       4     STABILIZATION *- -  OF CHEMICAL BONDS BY AROMATICITY
88       4     STRAIN EFFECTS *
86       4     STRAIN ENERGY CALCULATIONS *MATHEMATICAL PROGRAMS - -
91       4     STRUCTURAL ADDITIVITY RELATIONS *
88       4     TESTS *CONSISTENCY - -
93       4     TRANSITION *ENTROPY OF - -
92       4     TRANSITION *HEAT OF - -  UNKNOWN
```

Figure 5. Selected index entries (sorted) (step 5b)

A coding scheme must convey the following information to the computer:

What is the extent of the original text being used to form one or more index entries?

Which words have been deleted from the original text? What words have been inserted in the original text?

Which subphrases of the entire phrase are to serve as entry points?

In accordance with these requirements the following rules are used:

Scan the text line delimited by brackets symbol by symbol and use the following recording rules:

1) When a left (right) grouping symbol is encountered use a left (right) paren in its place.
2) When an undeleted word of the original text is encountered do not copy it, but copy its corresponding word-number supplied in the word-number line below it.
3) When rule 2) applies successively omit the word number, unless followed by a changed category.
4) When a deleted string of words is encountered use a single dash in its place.
5) When one (several) inserted word (s) is (are) encountered copy it (them) onto the coding sheet.

The rules given above are sufficient to create BICEPT statements. However, a more sophisticated coder can abbreviate the process by using a modified rule 2:

2') Only copy word-numbers of words that are immediately to the:
   a) Right (left) of a left (right) bracket
   b) Right (left) of a left (right) grouping symbol
   c) Left of an inserted word string
   d) Right and left of a deleted word string
6) Word numbers in BICEPT statements not otherwise separated must be separated by commas. Words within added phrases are automatically separated.
7) Spaces may be used at will in the writing of BICEPT statements, as they are ignored by the computer (except between words within added phrases).
8) A parenthesis-free BICEPT statement will yield a single index entry in text order, including any insertions or deletions indicated.
9) A statement involving only character strings or only text word tags is a valid index entry.
10) The number of BICEPT statements applying to one reference number is not less than the number of bracket pairs in the sentence. These may all be placed on one record (or card set) provided all statements except the first are preceded by a virgule.

RECORD FIELD DEFINITIONS

It may prove convenient to write the BICEPT statements onto standard coding sheets in preparation for keyboarding. Presently BICEPT statements are entered on punched cards. However, the wording below is chosen to apply whether cards, paper tape, or remote consoles are used for computer input.

Index entries associated with different reference numbers (sentences) appear on separate records.

BICEPT statements associated with the same reference number are separated by a virgule or are given in separate BICEPT statements.

Reference (Sentence) Field Columns 1 through 5. Write the reference number for the index entry in this field right justified and without leading zeros.

Continuation Sequence Field Column 6. If cards are used and a BICEPT statement cannot be completed on the first card, go to a second card, repeat the reference number, put a 1 in column 6 and continue the statement in column 7. If a third card is needed, put a 2 in column 6, etc. This convention provides a means of checking for proper card order within sentence.

If a BICEPT statement is blank *both* in the reference (sentence) field and in the sequence (trailer card) field, the computer can be programmed to supply the next following sentence number. Alternatively, these fields can be manually punched (without necessarily being written on the coding sheet) and the computer instructed to perform a card sequence check. If punched paper tape or an on-line console is the input, the first option appears preferable.

BICEPT Statement Field Card, columns 7 through 80; until the end record symbol in other cases.

INDEXING A NON-MACHINE-READABLE TEXT

This system may be used for a non-machine-readable text by making each BICEPT statement consist exclusively of user inserted phrases with no references to word-numbers, but otherwise unchanged. The user is still taking advantage of the machine's sorting ability, the feature of the BICEPT language that creates multiple index entries from a single phrase, and such other conveniences as listed under future developments below as prove useful. There is a distinct trend at present to employ computers at some point in the indexing cycle (*8*) and the capabilities of BICEPT should accelerate this trend, whether or not, as contemplated here, indexing is done from full text and irrespective of the fullness of the resulting index.

PERSONNEL REQUIREMENTS

Although there is much manual effort in this system at this stage of development, the professional indexer is responsible only for marking the listing, and for successive decisional steps as the index takes shape. The remainder of the work is clerical. If desired the BICEPT statements can be keyboarded directly from the marked listing, eliminating the coding—in effect combining the coder and keyboard operator.

Indeed, if a remote control station with cathode ray tube were used, the machine-produced tagged output might well be dispensed with and the text brought sentence by sentence on to the display tube. The indexing could proceed by keying in the appropriate BICEPT statements, one by one. While we have not done this, there seems no real difficulty in having the index entry formed from the displayed sentence as each symbol is introduced. The whole process would be completely intuitive.

PREVIOUS APPROACHES

Artandi (*13*), in her thesis, employed an authority list, a letter count, and presence of capitalization for detection of indexable concepts. Borko (*14*) and we in our previous work (*15, 20*) matched the text against a rejection list in the spirit of KWIC indexing, producing a coordinate type index. The method has the merit of being almost (or optionally entirely) automatic, but the drawbacks that manual review of the output is difficult, and the final index is not in conventional form. Nevertheless, the method would appear worthy of consideration where very full indexing such as in a manuscript index is tolerable or even desirable, and resources are not available to provide the manual processing involved in the current version of BICEPT.

As others before us have learned, the work on the earlier system showed that words out of context are not easily screened by humans. In essence, BICEPT is simply a method by which human indexing can be made as efficient as possible *both* in its clerical and in its decisional phases. For this to occur, the initial decisions *must* be made in the full text context.

FUTURE DEVELOPMENTS

As a special form of information processing, computer assisted primary index preparation necessarily involves (1) data (i.e., instructions—here BICEPT statements—and, optionally, such material as text to be indexed, a dictionary of common words, a list of generic or synonymous index entries, a list of lists of specific words to

be used as generic or synonymous entries, an authority list, and perhaps others); (2) input; (3) decision making; (4) operations; and (5) output.

It is here assumed that input of text results from a primary application, that preparation of dictionary entries and of alternants to text words and phrases is done once and for all, and hence does not constitute an input burden. Output is expected to be into a computer typesetting routine and, hence, likewise not chargeable. The one form of input chargeable to a specific indexing task is that of the appropriate BICEPT statements. Decision making is a human operation and hence, intrinsically outside of the BICEPT concept. Indeed, future research in primary index production will constitute, to a large extent, the shifting of this burden from the indexer to the machine. Were this possible completely at the present time, there would be no need of writing or keyboarding BICEPT statements. But while some progress is likely as time passes, fully automatic primary index production is presumably a long way in the future.

Several straightforward program features have been deferred in the current version of BICEPT purely because of the programming effort required. The simplest involves the provision of a battery of computer checking capabilities on incoming BICEPT statements. Statements need not be executed in sequence, but if inputted in sequence, a sequence check will detect a missing statement. If card input is used, cards of one BICEPT statement must occur in sequence.

To relieve the coder's difficulties of miscounting when a number of parentheses are to appear in juxtaposition, it would be possible instead to record a plus sign, +, followed by the parenthesis count, followed by the type of parenthesis involved. Thus three beginning parentheses could be symbolized as +3 (or five closing parentheses as +5). In interpreting a BICEPT statement the plus sign would alert the computer to set a counter to the following number, and then repeat the following symbol (the parenthesis) until the counter goes to zero. In checking a BICEPT statement for legitimacy by the computer, each opening parenthesis would be counted as a unit, each number following a plus would be counted for its value and the parenthesis following it ignored. Closing parentheses would be counted similarly.

At present no provision has been programmed to perform editing steps on BICEPT-formed index entries. Thus, all but one of the index entries on page 14 come from the computer ending with an unwanted "in." It cannot be deleted from the original BICEPT statement, however, because it is needed in that one index entry. Deletions, additions (for example to correct misspelling), and rearrangements are straightforward editing operations.

In the current version of the language only one level of qualification of the index heading is possible. If for a single index heading there is a large number of qualifiers which are themselves decomposable, then it may be desirable to provide additional levels.

For example, in the index of *Centrifuges in Cancer Research* there is an entry—Vibration, Flexural, of Freely Suspended Rotor—which contains two levels of qualification. The BICEPT language could be extended to handle this case through an additional set of delimiters—for example, < and > .

In a previous study (*16*) it was shown that any structure (in our case, any index entry) can be denoted by a code. There are two forms of codes, enumerative and analytical. The latter is a juxtaposition of a number of the former. The analytical code gains meaningfulness (i.e., structural representation) at the cost of length in number of symbols required. Economy in message handling (here index entry formation) calls for enumerative codes. For this purpose we use a reference tag consisting of sentence and word within sentence numbers. In the index as printed these are replaced by page (or section) and subpage designators.

While this works well for those words already in the text, and, indeed, in the sentence being indexed, the present version of BICEPT involves no economy if the word is outside the text or even outside the given sentence. Here again the solution is straightforward for words outside the text altogether. It consists in forming a list of such words together with enumerative tags by which the indexer would call for their full-form insertion into index entries. While such "key stroke economizing" would be of little benefit and even confusing to an amateur indexer, it could well contribute to, say, doubling the production of a trained professional indexer. A similar device, explained elsewhere (*17*) could be used to form an equivalent list from the words arising from the text being indexed. In this way any given word would be added in full only once.

The extensions described in this paragraph are the only ones involving true programming research and not simply availability of programming resources. Like other software, BICEPT is distinguished by the particular assortment of operations it facilitates. No method exists at present for locating and exploiting referents of anaphoric expressions in the sentence being processed (*18*). Should research on this problem be successful, presumably BICEPT language could be extended to take advantage of such results, or perhaps the results can be provided automatically. At present no provision exists for altering the order of appearance of words within the heading or within the modification of an index entry. The addition of this operation should not be difficult. Also, no provision exists for allowing the user to define and then exploit particular operations arising in special applications. It sometimes occurs that one form of the word (say the plural or verb occurs in the text, whereas another form (the singular or noun) is desired in the index. Presumably this too can be provided by adding list processing capabilities. Finally, phrase delineating techniques are under study (*19*), which, if applied prior to the tagged text printout stage, might markedly facilitate the human indexer's work.

ACKNOWLEDGMENT

The authors wish to extend thanks to A. Bondi for permission to use his paper in their studies, to J. H. Kuney and S. W. Walcavich and the Editor of the JOURNAL OF CHEMICAL DOCUMENTATION for making the typesetting tape available, and to Doris Williams for assistance in programming.

LITERATURE CITED

(1) *The Indexer*, 9 Pensioners Court, Charterhouse, London E. C. 1.
(2) Maloney, C. J., "Practical Preparation of Internal Indexes," *Indexer* **5**, 81–90 (1966).
(3) Davenport, W. C., and J. T. Dickman, "Computer-based Composition at Chemical Abstracts Service," p. 9, paper presented before the Division of Chemical Literature, 151st Meeting, ACS, Pittsburgh, Pa., March 1966.
(4) Baxendale, Phyllis, "Content Analysis, Specification, and Control," in "Annual Review of Information Science and Technology," Carlos A. Cuadra, Ed., Vol. **I**, Chap. 4, Wiley, New York, 1966.
(5) Newman, S. M., "Storage and Retrieval of Contents of Technical Literature, Non-Chemical Information," 2nd Supplementary Report, U.S. Patent Office, Washington, D. C., 1958.
(6) Yngve, V. H., "The Feasibility of Machine Searching of English Texts," International Conference on Scientific Information, National Academy of Sciences, Washington, D. C., 1959.
(7) Sedelow, S., and W. Sedelow, "Stylistic Analysis," in "Storage and Retrieval of Contents of Technical Literature, Non-Chemical Information," *op. cit.*
(8) Walker, J. F., and R. F. Schirmer, "The Indexing of Technical Books," J. CHEM. DOC. **6**, 26–30 (1966).
(9) "Annual Review of Information Science and Technology," *op. cit.* p. 389.
(10) Maloney, C. J., and M. N. Epstein, "Progress in Internal Indexing," *Proc. Am. Document. Inst. Ann. Meeting* **1966**, pp. 57-62.
(11) Bondi, A., "On Error Prevention," J. CHEM. DOC. **6**, 137–142 (1966).
(12) Hines, T. C., and J. L. Harris, "Computer Filing of Index, Bibliographic, and Catalog Entries," p. 126, Bro-Dart Foundation, Newark, N. J., 1966.
(13) Artandi, Susan, "Book Indexing by Computer," Ph.D. thesis, p. 207, Rutgers—The State University, New Brunswick, N. J., 1963.
(14) Borko, Harold, Ed., "Automatic Language Processing," Wiley, New York, in press.
(15) Maloney, C. J., James Dukes, and Sterling Green, "Indexing Reports by Computer," in "Technical Preconditions for Retrieval Center Operations," Benjamin Cheydleur, Ed., p. 13-28, Spartan Books, Washington, D. C., 1965.
(16) Maloney, C. J., "Semantic Information," *Am. Document.* **13**, 276–287 (1962).
(17) Glickert, Peter, "A Codification of English Words," The Author, p. P5, 1966.
(18) Olney, J. C., and D. L. Londe, "Language Processing. Anaphoric and Discourse Analysis," p. 4, unpublished paper.
(19) Clarke, D. C., and R. E. Wall, "An Economical Program for Limited Parsing of English," AFIPS Conference Proceedings, **27**, Spartan Books, Washington, D. C., p. 307–316, 1965.
(20) Beveridge, Gerald, and C. J. Maloney, "The Biological Laboratories Information Retrieval Program," p. 99, available from Office of Technical Services (AD 277 544), June 1962.

Keyboarding Chemical Information*

R. G. HEFNER, P. M. KEESECKER, and D. F. RULE
Chemical Abstracts Service, The Ohio State University, Columbus, Ohio 43210

Received September 21, 1967

During 1967, the Chemical Abstracts Service (CAS) will print some 570 million characters in its publications. A computer-supported keyboarding system has been developed at CAS to handle the data input process. The varied character set, approximately 1500 different type pieces, is being entered into the system through standard keypunch and typewriter keyboards. Input conventions have been developed following extensive analysis of character frequency counts. Substantial use is made of input formatting, shortcut techniques, and recognition of routine grammatical construction and of the natural structure of the data in providing inbuilt signals to the computer to allow automatic case and face changes. These techniques supplemented by a simplified keyed-flagging and a mnemonic code system are described in this paper.

The printed information services offered by Chemical Abstracts Service will, in 1967, comprise 131,000 pages carrying 570 million characters; the present rate of chemical information growth indicates that by 1970 CAS will publish 174,000 printed pages containing 760 million characters exclusive of any new services. To provide chemists and chemical engineers with ready access to this store of information, CAS is converting to computer base and will make information stored in computer files available not only for producing printed information tools, but also for direct searching. This conversion, which has recently been described in papers by Davenport (*1*) and Tate (*2*), requires that the most efficient means be developed for translating chemical information into machine language by keyboard devices assisted by appropriate computer programs.

Some 1500 different type pieces, or characters, are required to represent the nonstructural material in

*Presented in part before the Division of Chemical Literature, 153rd Meeting, ACS, Miami Beach, Fla., April 1967.