6. Output media
7. Running time
8. Decimal places required for accuracy

c. Errors. Error stops or other stops are listed with reference to the appropriate section of the flow chart. The listing also tells what should be done in case of an error stop.

IV. Flow Chart— Problem-oriented rather than computer-oriented and using a standard set of symbols.

V. The Program Itself. This section consists of the following items:

a. Language and program steps: a listing of the complete program for at least one computer

b. Cross references from one program step to other steps not in sequence

c. Operating instructions including plugboard wiring diagrams where necessary

VI. Sample Problems. Included in the sample problems are the required input data and the desired output format showing the results of the problem.

VII. Notation. This is a list of the nomenclature used in the text and of the symbols in the source program listing.

VIII. Literature References.

IX. Illustrations. Drawings and other illustrative material, prepared in a manner to insure good reproduction, is included in this section.

Editor's Comment: While the ACM has never adopted publication standards, it now appears that *program publication standards* for the ACM may be desirable in the near future. In my opinion, not only is the interest in program interchange growing constantly, but computer-independent language development is bringing us closer to the day when widespread publication of truly catholic programs will be an important function of computer-oriented societies.

Mr. Kent's abridgment of the AIChE committee's report avoids machine-oriented details such as punched card formats; it outlines a procedure which is essentially machine-independent, yet can be used for machine-language publication.

The full report describes means by which program publication is announced and plans for distributing and for covering the cost of small-scale publication by the Society. H. S. B.

A Proposal for Character Code Compatibility

R. W. Bemer, *I.B.M. Corporation, White Plains, N. Y.*

The emergence of a single standard from a welter of conflicting precedents depends upon two solutions:

1. selection or development of an adequate and logical standard,

2. phasing out (or peaceful coexistence with) the old varieties.

This paper deals with the latter problem and proposes the mechanics for a solution in the area of character codes, as represented by bit combinations.

It appears impossible to reconcile the many different codes in use on paper or magnetic tape such that a particular code could be the national or international standard. Because of the wide usage of these various codes they must be considered parallel standards subject to atrophy through adoption of a single superior code. A simple device that I call the "escape" character will allow as many compatible and graded standards as there are bit combinations in any number of tracks, although it is certainly not desirable to have more of these than absolutely necessary.

Given T character tracks (not feed, parity, or control tracks), there are 2^T possible code combinations. Normally these are all assigned to specific characters or controls. I propose that *one* of these combinations, the *same* one for all standards, be reserved as an "escape" character. This is to be excluded from every such set of characters assigned.

Regarding the choice of this character, it is unwise to use a *null*, or absence of punches or bits. Furthermore, it is quite possible that the physical permutation of tracks on tape will not be in direct correspondence with the bit pattern of internal storage in a computer or data-processing device. The only code that avoids these difficulties is the completely punched combination, or all *ones* in the bit structure.

Let us make provision for this "escape" combination to interrupt normal decoding of a stream of characters. It will say, in effect, that "The next T-bit combination is to be considered a numeric identifier of a particular standard." From then on, until interrupted by an "escape" character in *that* set, all combinational T-bit characters will be interpreted according to that standard. Shifting from one standard to another is therefore *dynamic*. A great additional advantage of such a scheme is that many messages in several different codes may be adjoined in the stream of transmission. In hardware, the "escape" character can be made to interrupt to set relays or other switching devices to select one of a variety of readers or decoders.

Another way to view this mechanism is as a two-character shift symbol, comprised of a common symbol and an identifier number. If such numbers are limited to exclude the "escape" combination itself, then two or more "escape" characters in sequence will permit a still further variety of standards (or, technically, sub-standards). This is a recursive property.

Admittedly there is some difficulty in applying this principle to most existing codes. For example, the Baudot 5-track code for paper tape already has the "escape" combination assigned to "Letter Shift." Fieldata code has this same drawback, at least in the August 1959 version, in that this code is assigned to the letter Z and to Backspace. Although the 5-track teletype codes are virtually impossible to reconcile in this manner, it is suggested that all future codes of 6 or more tracks carry this "escape" provision as a built-in safeguard against obsolescence.

A Terminology Proposal

Fred Gruenberger, *The RAND Corporation, Santa Monica, California*

It has been suggested that we need some new words in our industry. For example, consider the following two definitions of the term *random access*:

(1) Access to storage under conditions in which the next position from which information is to be obtained is *in no way* dependent on the previous one.

(2) Feature of certain internal memory systems, *particularly magnetic drum type*.

(In both cases the italics are the author's.)

Definition (1) is taken from the last official ACM glossary (*Communications, ACM*, October 1958). It has been reproduced verbatim in numerous other glossaries. It is probably the most widely accepted definition of the term today.

Definition (2) is from a booklet titled "What Every Businessman Should Know About Electronic Brains," published by a prominent manufacturer of computers. The subtitle of this booklet is "Or Facts That Will Make You a Cocktail Party Expert on Office Automation." This is a surprisingly accurate title.

This term *random access* is one which is getting kicked around a bit too loosely. One sees phrases like "our new disk file is a random access device with an *average access time* of . . .". If definition (1) is correct, such phrases are internally inconsistent.

It is a fair guess that nearly everyone would agree that tape storage is not random access. But we lack a suitable antonym. Further, those who are pushing disk files are eager for a trichotomy, on the grounds that disk storage is more random than tapes, albeit less random than cores (as used in the 704). The locution "more random" is a logical monstrosity, akin to "more pregnant" or "more unique."

We cannot hope for watertight divisions here. For example, it has been pointed out that, in strict adherence to definition (1), the core storage in Stretch is nonrandom. Yet it is a long way from the time-dependent access one finds in a disk file.

Therefore, the use of the following three terms is suggested:

(1) Constant-time access
(2) Variable-time access
(3) Serial access

Constant-time access would apply to storage in vacuum tubes, cathode ray tubes, magnetic cores, rods, Biax elements, and the like. Variable-time access would apply to drums, disk files, tape bins, carrousel storages, and so forth. Serial access, of course, applies to tapes.

"Constant" is preferred to "fixed" to include stores like that on the 604, where the time of access is a function of the oscillator frequency at the moment. Constant-time access replaces exactly the term random access as defined in (1) above.

The three terms used in this proposal were devised by Irwin Greenwald.

Editor's Comment: If we specify that "access" is to a single word location, and if we ignore the queuing-time variability of partitioned, buffered stores, Fred's examples imply that "Constant-Time Access" word selection involves only the direct application of digital address information (e.g., switching), *at microsecond speeds*. (We must stretch a concept and call cathode-ray beam positioning "switching".) Unfortunately, some slow stores (e.g., one-revolution-per-word multiple-interlace drums) involve essentially *constant* access time, while our term should imply *short* access time. Perhaps "Prompt Access" would be more lucid.

His "Variable-Time Access" involves at-least-partially-analog address information (e.g., rotating device angular position, moving head location, acoustic pulse position in a stream, etc.).

"Serial Access" would seem to be a special case of "Variable-Time Access", for which the selection in sequence of *any two* different word locations requires the passing over of *all* intervening address locations. H. S. B.