Letters to the Editor

An Open Letter to Bob Bemer

Dear Mr. Bemer:

As far back as Fall of 1958 I recall your mentioning that if ALGOL were not developed as rapidly as possible, FORTRAN would become a standard in the industry by default. At that time, just as I was preparing a FORTRAN compiler for the Datatron 205, several others were preparing similar compilers for various other machines. Even today FORTRAN is being written for several machines, and Philco recently announced a compiler for their machine which will accept FORTRAN as a compatible language.

ALGOL has been with us in spirit for some time now, but that's about all. There are a few exceptions but none of any significance. Could it be that the compiler builders have forgotten that the rest of the industry is still writing in FORTRAN or some less sophisticated technique, and that the amount of work being done this way increases every day? Also, if we wait much longer for ALGOL, whatever standardization effects it once offered will be completely lost as others follow the recently announced policy of General Electric in creating or attempting to create their own versions of a super, general compiler. At the 1959 ACM National Meeting and again at the 1960 Western Joint Computer Conference, you indicated that IBM has both ALGOL 58 and ALGOL 60 running. If it is the feeling of IBM that they do not wish to be accused of dominating the industry in the selection of a new "standard" and therefore they will wait for the ACM or someone else to make this selection, then in my opinion it is the wrong attitude for them to take. As the largest manufacturer with the most machines in the field and therefore the most users, and also the most successful compiler builders, IBM has the right to exert its position. Philco has proven that it is reasonable to produce new compilers with the ability to accept previous languages so that old programs need not be lost. Also the industry has shown that it needs the strong leadership that IBM can provide in this area, for without this leadership a select few fiddle while the industry burns!

The articles published in the ACM *Communications* and other places are sometimes of academic value, but they do nothing to alleviate the urgent need which exists for a more sophisticated compiler than FORTRAN. Let the compiler builders argue over the desirabilities of the left-to-right scan over the right-to-left scan, or the two-operator triple over the two-operand triple, but give us ALGOL now. Pick the version which you like the most and release it. What the computing public wants to read in the *Communications* is that a new and better compiler is here for them to use now. Articles that tell of an insignificant compiler that serves a few people at one installation do not make very good reading. These compilers are a considerable compromise when one considers what they might be today if IBM had continued to lead the industry as they had been, two years ago. At that time it was generally indicated that the other manufacturers were ready to follow IBM's lead in standardizing their work. It was my impression that IBM would lose its position as leader only through the supreme efforts of others and not by suddenly coming to a standstill and letting the rest of the world drift by.

Lets have some action.

RENE DE LA BRIANDAIS, *President*
Digital Computing Services
P.O. Box 13
Pinole, California

Re: Rene De La Briandais' Letter on FORTRAN

Dear Editor:

Although Rene has some well-taken points in his letter, he has not emphasized some of the other sides of this very complex problem. I should like to point out some of these other facets to help the membership gain insight into an often confused situation.

1. Rene implies throughout his letter that there has been no progress made in either use or capacity of production algebraic compilers since the 1956 introduction of FORTRAN. This is not so. Both the FORTRAN language and compiler have been constantly revised. Diagnostic editing and separate assembly program facilities have been added. Although ALGOL is admittedly a superior language (it should be, for IBM's own FORTRAN and experimental languages made heavy contributions), FORTRAN is the present workhorse and is already operative in a large number of installations and understood by thousands of people. It would be unwise to give the user language elegance and take away productivity and efficiency.

2. Under present circumstances it is difficult for ALGOL to offer the standardization effects Rene mentions. Two major versions of ALGOL exist—1958 and 1960. Both of these have been chosen as bases for various processors and the confusion is further compounded by a number of dialects. I cannot agree that ALGOL in its present form can be a standard as long as Joe Wegstein maintains his ACM Committee on ALGOL Maintenance and Changes.

3. For some of the same reasons, I see difficulties that prevent FORTRAN from becoming an industry standard.

4. How refreshing for Rene to ask that IBM exert its position! I remember the time when the common cry was the other way around. We are past the time of unilateral decisions in data processing standards. As you know, the X3 Sectional Committee of the American Standards Association is now the prime focal point for this problem. We realize that the user wishes to have a part in the formulation of standards as well as the supplier. Since we do not know what the standard algebraic, or perhaps combined scientific-business, language will be, we are attempting to invest our programming talent most efficiently by creating highly generalized processors with facilities to generate translators from language definition.

5. Rene asks us to give him ALGOL now in place of FORTRAN. Does he wish to do without the input-output facilities and operating system of FORTRAN? I think he places an undue emphasis upon the procedure language and ignores all the fine advances IBM and other companies have made in the production of such vital programming tools as input-output control systems, operating systems, sort and merge generators, report and file maintenance generators, diagnostic routines, and even generalized applications. If IBM has not released a series of ALGOL processors for its various machines, it is perhaps because it's our policy to make progress in large, discrete jumps. There is more to a processor than the language. We are constantly in search of new production methods. We realize that we cannot simply use n times as many programmers for the new machines that will be n times faster. We are quite aware of experimental new techniques, many of which have been developed at universities. But we must maintain high production standards for the customer and this involves time. Let us remember that the principles for achieving atomic power were developed in 1938 but the practical production power stations did not come for another 20 years.

When there exists a language fairly safe from arbitrary change and when both the language and processors offer enough further advantages to customers to offset the costs of re-education, programming modification, and general dislocation—then we will issue a new system with which the user may choose to supplant FORTRAN.

6. I hope the articles I have published in the *Communications* have stimulated interest and thought among the production programmers in this country. Hundreds of compiler programmers cannot be converted overnight to the latest symbol manipulation techniques of the universities without disrupting production. If production is disrupted there will be hundreds of Letters to the Editor about broken promises.

7. Despite the escape clause of the "reference language", ALGOL will not really be usable until new input-output equipment exists which will handle the character set directly. This area is under experimental investigation, and the production of acceptable new hardware takes considerable time.

8. Rene asks both that IBM provide "strong leadership" and also "exert its position" as a right. Such a dual role is impossible. I suggest that he read the American Standards Association Document PR27, which describes the operating methods by which standards are achieved. He will find that standards are voluntary and only have force when embodied in specific law. Furthermore, to become standards they must have the approval of users, manufacturers and general interest groups.

R. W. BEMER

Word Inversion

Dear Editor:

The method I described for word inversion on the IBM 709 (*Communications*, December 1960, page 658) suffered from some "inverted thinking" on my part. The following paragraph corrects the description of the CAQ table used and should replace the last paragraph of my original contribution:

The appropriate octal word for the ith table value is n = XX0000000000, where XX represents the inverse of i as defined by Price in Method 1.

PAUL E. DES JARDINS
Aero-Space Laboratories
Missile Division
Downey, California

Round-Off

Dear Editor:

The letter by Roland Silver on the subject of round-off published in the December 1960 issue of the *Communications* prompts me to point out that the Bendix G-20 computer contains a hardware round-off rule essentially the same as Silver's method number 3. The G-20 General Reference Manual states:

"Round-off where called for is always accomplished as follows: if the discarded portion is less than one half, it is ignored; if it is greater than one half, the least significant digit saved is increased to the next higher unit; if it is exactly one half, the last digit saved is rounded to the nearer odd value. This is analogous to the rule in decimal arithmetic which rounds to the nearer even digit. Its purpose is to eliminate bias and to reduce "noise" in the case of repeated rounding of the same number."

C. A. PIPER, *Manager*
Applied Mathematics
Bendix Computer Division
Los Angeles 45, California

On the Calculation of Interest

Dear Editor:

I have read the note by Ingerman, on the Calculation of Interest, in the October number of the *Communications of the ACM*. This contains an algorithm which appears not to be in standard ALGOL 60. In particular:

(i) there are no specifications corresponding to the formal parameters;
(ii) the first **"begin"** should follow the procedure declaration;
(iii) semi-colons at the ends of the printed lines have been omitted;
(iv) asterisks have been used instead of multiplication signs;
(v) there seems to be an error in an "if" statement near the end:

```
if p ≠ ib then
  if Tp > ap then
  begin Tib := Tib + Tp − bp ;
    Tp : bp end
```

Should, I think, read:

```
if p ≠ ib then
  begin if Tp > ap then
    begin Tib := Tib + Tp − bp ;
      Tp := bp end end
```

(vi) there are two misprints: in one place "to 1" is printed for "to1" and in another "then" should be in bold-face type.

You are almost certainly aware of these points, and you may well have commented on them in a more recent issue of the *Communications*, which I have not yet received. However, my point is that I feel that algorithms appearing within papers should follow the same excellent standards of the Algorithms Department.

F. G. DUNCAN
The English Electric Company Limited

Northwest Computing Association

Dear Editor:

With reference to C. M. Sidlo's letter in the December issue concerning computing groups, users' and otherwise, I should like to state, briefly, the raison d'être and aims of the Northwest Computing Association which he cites in his letter.

The Association is open for membership to anyone who wishes to join, and attempts to cater for all types of computer persons. A meeting is held in the Seattle area once a month at which a speaker or speakers deal with some aspect of computing, be it from the user's point of view or the vendor's, scientific or non-scientific. It also holds a one-day conference in the spring at which most of the speakers are from the northwest, and a two-day conference in the summer to which speakers and delegates are attracted from all over the United States and Canada, and at which vendors display their wares. Missionary work is also conducted, with zeal, amongst high schools and junior colleges.

The object of these exercises is to provide a forum for the interchange of ideas and information amongst computer people living in a remote and inaccessible area—a professional surrogate. The local definition of a professional programmer is he who, having discovered the existence of ACM, ALGOL and creeping polymorphism, begins to dispatch applications southward to the Promised Land. This normally happens two years after he was hired as a greenhorn and when he is beginning to deliver the goods. The Columbia River is a semi-permeable membrane and the N.C.A. was inaugurated to reduce the osmotic pressure.

This in no way supplants users' organizations, and I see no indication of any demise of SHARE or USE. Rather, N.C.A. acts as a focal point for general discussion by people with varying backgrounds, interests and machines—an adjunct to the users' organizations.

NORMAN SANDERS, *President*
Northwest Computing Association
Box 838, Seahurst, Wash.