

高専土木科学生を対象としたFÖRTRANプログラミング教育

大阪府立高専 正会員 安東 祐一

1. はじめに

昨年のシンポジウムにおいて高専土木科における電算機利用教育の概要について発表した。電算機利用教育の目的は、“電算機の能力を過不足なく正しく認識させ、その有効的な利用法を考えさせること”であると筆者は考えるが、そのための手段として、1). なぜプログラミング教育が必要であるのか、2). なぜ言語として、FÖRTRANを採用するのか、について始めに述べる。つまり、最近その充実の度合が著しいFÖRTRANの文法のうち、どの機能を捨て、どれを捨てるのかについて述べる。

2. なぜプログラミング教育が必要であるのか

学生が電算機に慣れ親しみ、さらに電算機の能力を過不足なく正しく認識するためには、どうしてもひとりで以上のプログラミング言語を知っておく必要がある。(もっとも、年ごとに漫習用に使えらる電算機がない場合には、あえてプログラミングを行なわねば、学生にコンピュータ・アレルギーを起さざるだけであろうから、その場合は話が別である。) アプリケーション・ウェアの目標のひとつである“電算機の知識がなくても使える”という時代がくるまでは、学校でプログラミング教育を行なわねばならない。

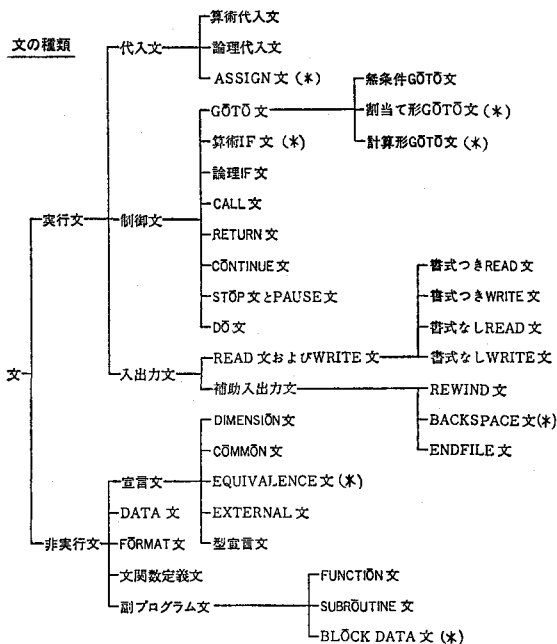
3. なぜ言語としてFÖRTRANを採用するのか

結論から先に述べると、採用する言語としては“普及度が高く、規格が制定されている言語”の方がよいからである。FÖRTRANが、非の打ち所がないすばらしい言語とは筆者も考えないが、普及しているということは、言語としての最大の強みである。FÖRTRAN以外にも良い言語はいくつかある。FÖRTRANと同じレベルで、PL/I、その上のレベルで問題向き言語(PÖL)、PÖLとまでいかなくても電算機メーカーで用意しているアプリケーション・ウェアに便利なものがある。これらの言語が、なかなか広まらないのは、普及していないからである。妙な理屈であるが、厳然たる事実である。たゞ、この考えで、学校においてFÖRTRANを採用し続けるならば、FÖRTRAN人はばかりが増し、他の言語は、いつまでたっても日の目をみないことになる。そこで、学校においてFÖRTRAN以外の言語を採用する場合を考えてみる。FÖRTRANは、算術論理の記述には適しているが、専門分野で問題の処理手順(アルゴリズム)を記述するには細かすぎる。専門知識と情報処理技術の相乗的な課題に取り組む場合に適した言語は、PÖLであろう。PÖLでは、計算の部分はブラック・ボックスになっているが、このブラック・ボックスをある程度信用して使っていかなければ、大きな問題には取り組めない。FÖRTRANという強力な言語をあえてあとにまわし、他の言語を採用するとすれば、上に述べた理由でPÖLしかない。現在、PÖLを、電算教育の最後の段階で、こういう便利な言語があると紹介しているが、漫習が可能ならば授業の最初から取り入れてもよい。電算機に慣れ親しみという電算教育の最初の段階では、FÖRTRANでも、PÖLでも同じであり、PÖLの方が、よほど親しみやすい。その後、学生が、ブラック・ボックスの中身を知りたいという場合に、そこで使われている言語を教えればよい。目的があれば理解も早いであろう。PÖLのうち、教育用に使えらるものとしては、ICESのサブシステムのコマンドしか考えられない。近くの大学の共同利用と

ンタで、このシステムを導入していただいて、リモート・バッチで使えるようになれば、FORTRANではなく、POL から入るという教育を行なってみようかと考えている。

4. JIS FORTRAN の3つのレベルについて

かつてFORTRANの入門コースで、3000レベルの文法を教えたこともあるが、大阪府立高専の小型の電算機（FACOM 230-28、CPUメモリ64KB）ですら、7000レベル以上の文法が使える今日では、3000レベルで教え始めることの意味がうすらいできたと考える。入門コースのテキストで、3000レベルを意識して、論理IF文を使うが算術IF文だけを押し通しているものを見かけるが初心者にとって、どちらのIF文の方が理解しやすいかということには、重きをおくべきではなからうか。3つのレベルの相違点は、文献1)、2) を見てもらうとして、7000レベルまでの文の種類を右に示す。(*)印をつけたいのは、授業で積極的に教えない（知っておく必要はあるが、あまり使うなと暗示をかけている）ものである。担当教官の好みと学生に押しつけているのかもしれない。



5. JIS FORTRAN 7000レベルまでの文法で学生が抵抗を示すところ

JIS FORTRAN 7000レベルまでの文法で、学生がどこでとまどうかは、エラーが起きて、その原因の相談にやってくる状況で把握できない。プログラミング教育を数理解当していると、その個所よりよき見当がつくので、授業中その個所ではていねいすぎるほど説明しているにもかかわらず、やはり間違える学生が出てくる。そういった個所を次に述べる。

1). 配列宣言を忘れたためのエラー

副プログラムで配列宣言を忘れる学生が多い。

2). 配列と、その宣言した大きさを超えて使用したためのエラー

コパイニシアル・フリーでない計算機では、エラーであるにもかかわらず、ちっとも正しい答が出たりする点は、やっかいである。学生に電算機不信をおこさせかねない。

3). FORMAT文と補記述子と入出力量の型の対応がつかないためのエラー

あまり複雑な入出力文、FORMAT文を用いると、この間違いをおかしやすいので、できるだけ分割するよう指導する。

4). 副プログラムの仮引数と実引数との対応関係

引数で受け渡すには、コモンで受け渡す方法は理解しやすいらしい。自分でプログラムを組

んでいる間は、わざわざコメントで受け替えているが、数値計算用、プロット用の書きあいのサブルーチンは、引数で受け替えるようになっているので、こゝでとまどう。整合配列の手法と、呼び出しプログラムで宣言された配列の手法が異なる場合の対応関係などは、教室で一方的に説明してみても、なかなか理解しない。Trial and Error で学生に理解させることにしている。

このような箇所は、別の見方をすれば、FORTRAN の改良すべき点といえるかもしれない。

6. JIS FORTRAN 7000 レベルにない機能の取り扱い

最近の FORTRAN は、JIS FORTRAN 7000 レベルにない機能を数多く有している。そのうち、まず、プログラミング教育の初期から取り入れるものを次にあげる。

- | | |
|-------------------------------|--|
| 1). 引用符で囲んだ形式の文字定数 | 例: 'ABCD' |
| 2). 3次元を越す配列 | : A(I, J, K, L, M, N) |
| 3). 異なった型との演算 | : P = Q * N + 3.3 + 2 |
| 4). READ 文の ERR および END パラメータ | : READ(5, 501, ERR = 8, END = 9) A, B, C |
| 5). 整数型、論理型を転送できる付随記述子 | : @14.7 |
| 6). リスト指示の入出力文 | : READ(5, *) I, A, B |
| 7). データ・セット参照番号を省略した入出力文 | : PRINT 601, I, A, B |

つぎに、プログラミング教育の後期で取り入れるものをあげる。

- | | |
|-------------------------------|----------------------------------|
| 1). 関数利用の実数値として使う文字定数 | 例: CALL PRNT('A = ', A) |
| 2). 任意選択用の長さの整数型、実数型、複素数型、論理型 | : REAL*8 |
| 3). 一般化された添字の形式 | : A(N * M + 0.5) = ... |
| 4). PAUSE 'メッセージ' | : PAUSE 'PAUSE 1' |
| 5). T 付随記述子 | : T101 |
| 6). IMPLICIT 文 | : IMPLICIT REAL(K) |
| 7). 型宣言文での長さの指定 | : REAL*8 MAXIMUM |
| 8). ENTRY 文 | : ENTRY PUT(X, L) |
| 9). デバッグのための機能 | : DEBUG TRACE |
| 10). FORTRAN 提供の関数の自動選択 | : GENERIC |
| 11). 記憶域内での書式によるデータの交換 | : ENCODE / DECODE |
| 12). 事務用付随記述子 | : \$IC10 |
| 13). データ・セットのパラメータの設定 | : FILES Z-MT(BL = 256, RL = 256) |
| 14). ELEMENT 文 | : ELEMENT FIBOBI |

入門コースでは初學者がより抵抗なく FORTRAN になじめるであろうと思われるもののみ採用した。後期で取り入れるものは、知っておくと便利だと思えるものである。

7. FORTRAN コンパイラの持つ諸機能の取り扱い

FORTRAN のコンパイル条件を、処理プログラム実行開始文に記述することができるが、そのうち FORTRAN にある程度なじみのある学生が、知っておいてもよいといふものがいくつかあるので、それらをプログラミング教育の最後で説明している。

- 1). プログラム・リストの出力の有無の指定
- 2). 最適化指定
- 3). プログラム全体にわたってのデバッグ指定
- 4). プログラム中で使用している定数、変数、関数の精度を変更する指示
- 5). プログラムの各行目に文字*が記述されているとき、その行を有効にするか無効にするか(注釈行とみなす)の指定

8. テキストおよび文法書

FORTRAN プログラミング教育のテキストとしては、自習書としても使えるような、できるだけいいに解説してあるものが望ましく、また廉価である方がよい。現在用いているのは、学校に導入している電算機のメーカーが出している FORTRAN の入門書であるが、これも最近価格が急によってきたので、メーカーの了解のもとに、リプリント版を学校で用意する予定である。文法書としては、メーカーで用意しているマニュアル類の中から、FORTRAN の最新の文法、組込み関数、基本外部関数、プロット、サブルーチンの文法、断断メッセージの解説を、こちらで編集しなおして1冊にまとめ、学生に持たせている。

9. おわりに

電算機利用教育は、その内容が新しい分野の技術であり、速い速度で変化しているので、常に最新の技術の動向を知り、それを完全に自分のものとして教育に取り入れる努力が必要である。今回は、FORTRAN プログラミング教育を取りあげたが、FORTRAN の文法も最近その充実の度合いが著しい。筆者が初めて FORTRAN に接した10年前のもの比べるとずいぶん便利になった。たとえば、算術式・関係式の記述において許されない組合せといふものがほとんどなくなったし、リスト指示の入出力文を用いることにより FORMAT 文から解放された。さらにデバッグのための機能が充実しつつある。担当教官としては、これらの機能を十分に理解したうえで適当に取捨選択しなければならない。FORTRAN を採用する姿勢とともに、これら FORTRAN の機能の取り扱いについて現状を報告した。批判があればありがたい。

(参考文献)

- 1). JIS ハンドブック 情報処理 1976 p.287-401 日本規格協会
- 2). JIS FORTRAN 入門(下) 森口繁一 p.74-98 東文出版会 (1970)
- 3). FACOM 230 FORTRAN IV 文法書 富士通 (1974)