

寄 書

JIS ALGOL に対するコメント*

佐久間 紘 一**

JIS ALGOL¹⁾ を読んで気がついた点がいくつかあるので、述べてみたい。

1. output 1 と input 1

output 1 (channel, ' ', E_1) と input 1 (channel, ' ', E_1) は、水準 50 と水準 60, 70 とで異なる。水準 60, 70 では、output 1, input 1 は outlist, inlist を使用して定義されていて書式の記号列が空の場合、'N' が必要なだけあるものとして実行する。すなわち、おのおのは

```
output 1 (channel, 'N',  $E_1$ )
```

```
input 1 (channel, 'N',  $E_1$ )
```

と同じことになり、前者は E_1 の値を標準書式で出力し、後者は標準書式で入力しその値を E_1 に代入する。

しかし、水準 50 では output 1, input 1 が通常の日本語で定義されており（水準 50 には outlist, inlist がないので）、書式の記号列が空の場合 'N' を補わず、その実行を完了するようになっている（水準 50.5.5）。結局、output 1 (channel, ' ', E_1) と input 1 (channel, ' ', E_1) は入出力を何も行なわないで実行を終る（channel の計算による副作用をのぞけば空文とおなじことになる）。

このように水準 50 と水準 60, 70 とで異なるのは、output 1, input 1 で書式の記号列が空か、表題の書式と字くばり記号だけ（どちらか片一方だけでも）の時に起こる。いいかえれば、数の書式か標準書式が入っていない時におこることになる。

2. 入力の標準書式の区切りと行の右端

入力の標準書式による数の右端の区切りは

(1) 小数点、符号、数字または 10 の右側に現われた字であって、小数点、符号、数字または 10 以外のもの。

(2) k 個の空白の列、 k 個より少ない場合は無視

する。

(3) 小数点、符号、数字または 10 に対応する記号が、数の始まりと同じ行に現われたとき、その行の右端。ただし、小数点、符号、数字または 10 が、その行に現われないときは、(1)か(2)の種類の区切りが見出されるまで、複数行にわたって読み込む（入出力、3.1.2）となっている。だから行の右端は区切り記号として使うこともできるし、読みとばされる場合もありうると考えられる。また実際水準 50 以上では (hend, vend を使わないかぎり) そのようになっている。しかし水準 30, 40 の inreal は、「区切り記号で右端を区切られた数の情報が発見されるまで $\rho+1, \rho+2, \dots$ を調らべる。…。これらの手順に際して、 ρ が行の限界 P をこえる場合には、その効果は規定しない」（水準 30, 40.5.2）となっている。こうなると行の右端を区切りとして使用したり、行の右端まで空白（その空白の数は k 個未満とする）にして次の行に数を書いておくとうなるのであろうか。

3. プログラムが複合文のときの名札の有効範囲

プログラムが複合文のとき名札の有効範囲に問題がある。つぎのプログラムを考えてみよう。

```

10: begin
    go to 12;
11: begin
    ...
    end;
    go to 13;
12: begin
    ...
    end;
    go to 11;
13:
    end

```

} ブロック 1
} ブロック 2

名札の有効範囲については「ある文の前にコロン(:)を隔てて置いた名札は、その文を含むもっとも内側のブロックの頭(block head)にさかのぼって宣言

* Comments on JIS ALGOL, by Koiti Sakuma (Kyoto University)

** 京大数理解析研究所

されたものとみなす。ただし、プログラムの最初の **begin** の前の名札の意味は規定しない(水準 7000 等. 4.1.2)と書いてある。よって l_0 の意味は規定されていないから、どうなってもよい。では l_1, l_2, l_3 の有効範囲はどこであろう。普通に考えれば、ブロック 1 とブロック 2 をのぞいたプログラム全体とおもわれる。しかし、たとえば l_1 に関していえば l_1 : **begin** ...**end** というブロック 1 を含むもっとも内側のブロックを考えなければならない。しかし、その外側にあるのは複合文だけであるから、そのようなブロックは存在しない。そうするとこのようなプログラムは書いてならないのであろうか。書けるようにする簡単な方法は、プログラムの外側にもう一つ仮想的なブロックを作り、そこで **abs** などの標準関数を宣言されているというふうなことであろう。しかし、そうすると l_0 が意味を持ってしまっただというのなら、「プログラムが複合文の場合には、その複合文をブロックとして扱う」という文を入れるのがよいと思われる。

(なおこのことは、JIS 以外の ALGOL²⁾ に対しても問題になる)

4. 名前変えの仮パラメタに対する型の制限

水準 7000 で規制つきの名前仮パラメタの型と実パラメタの型が一致しなければならないのかどうか、よくわからない。どうも一致しなければならないように読めてくるし(水準 7000. 4.7.4(5)), 解説の 3.1 に

は水準 7000 は国際標準規格水準 0 に対し「名前変えの仮パラメタに対する型の制限で異なる」と書いてある。しかし、水準 7000 の解説の例 5 の中では、**real** という規制を持った仮パラメタ F が、 $J \uparrow 2 + K \uparrow 2$ (J も K も整数型) という整数型の実パラメタに対応している。

5. □ の金物での表現

解説の基本記号の表わし方で、基準言語での '□' の金物での表現を「'□' (記号 □ は、+ や - や空白を用いることを表わす)」としている。どうして、ただの空白ではいけないのであろうか。たとえば基準言語での '3□.□1□□4□' を金物での表現でそのまま '3□.□1□□4□' とせず '3.□..□1.□.□4.□' とするのであろうか。ただの空白としても、他の金物での表現と矛盾のないものができると思うがどうしてこういう表現を例としたのであろうか。

参考文献

- 1) 電子計算機プログラム用言語 ALGOL (水準 7000, 6000, 5000, 4000, 3000), 電子計算機プログラム用言語 ALGOL の入出力 (水準 70, 60, 50, 40, 30), 日本規格協会, 1967
- 2) P. Naur: Revised Report on the Algorithmic Language ALGOL 60, Comm. ACM Vol. 6, pp. 1~17, 1963

(昭和 45 年 3 月 19 日受付)

○「情報科学と統計学」シンポジウム

日本学術会議の長期研究計画委員会および統計学研究連絡委員会では下記により「情報科学と統計学」シンポジウムを開きますので、お知らせいたします。

日時: 45 年 8 月 5 日 (水) 10: 30~15: 00

会場: 日本学術会議大会議室 (港区六本木 7-22-34)

次第:

司会 森田優三(福岡大学)
松下嘉米男(統数研)

- (1) 開会の辞 統計学研究連絡委員会委員長
北川敏男(九大)
- (2) 数学と情報科学について
国沢清典(東工大)
- (3) 統計数理と情報科学について
林知己夫(統数研)
- (4) 情報工学について
坂井利之(京大)

(5) 基礎工学と情報科学について

田中幸吉(阪大)

(6) 総合討論

(7) 閉会の辞

小野周(東大)

「情報処理月例会(8月)」予告

8月18日(火) 15: 00~17: 00 に機械振興会館66号室(6階)で「運輸省自動車登録・車検システムのソフトウェア」について、津田宏明氏(電々公社)が講演いたします。

論文賞の創設について

去る4月28日開催の第8回通常総会における本年度事業計画で明らかにしましたように、10周年記念事業の一環として、本誌既載の懸賞論文賞のほかに、本年度から下記「表彰規程」に基づいて、論文賞を創設いたしましたので、お知らせいたします。