

流れ図付きソースプログラム作表システム： Fortflowの改訂について

牛島, 和夫
九州大学工学部情報工学科

藤村, 直美
九州大学情報処理教育センター

掛下, 哲郎
九州大学工学部情報工学科

<https://doi.org/10.15017/1470369>

出版情報：九州大学大型計算機センター広報. 18 (1), pp. 38-41, 1985-01-25. 九州大学大型計算機センター

バージョン：

権利関係：

流れ図付きソースプログラム作表システム — Fortflowの改訂について —

牛島 和夫*, 藤村 直美**, 掛下 哲郎*

1. はじめに

プログラムの制御の流れを明確にすることは、その文書性を高めたり、テストを支援するために有用である。しかしFortranプログラムは制御構造が貧弱で、GOTO文を多用しがちなため制御の流れが複雑になりやすい。このような場合の対策として、流れ図を作成する方法がある。ソースプログラムを読んでその流れ図を自動的に作成しプロッタやグラフィックディスプレイの画面上に作図するシステムはかなり出回って使用されている。その図をラインプリンタ上に出力するものもある。ところでこうして作られた流れ図ともとのソーステキストとを対応付けようとする、その作業は必ずしも容易であるとは言い難い。

九州大学大型計算機センターでは、Fortranソーステキストの左側の余白を利用して流れ線を印刷するプログラムFortflow [1] が使用できる。同センターではこのたびFortranをFortran 77に統一することになった。それに伴いFortflowをFortran 77に対応させるための改訂作業を行った。

2. システムの概要

新しいFortflowによる出力を図1に示す。このシステムは既存のFortflowに以下の機能を追加したものである。

(1) ブロックIFの構造の対応を示すために、同じIFレベルをもつブロックIFの構造を作る文(ブロックIF文, ELSEIF文, ELSE文, ENDIF文)を歯状につなぐ。さらに、そのブロックIF全体の先頭を目立たせ、流れ線を見ただけでプログラムの構造がわかるようにしている。

* 九州大学工学部情報工学科, **九州大学情報処理教育センター

(2) ブロック I F 文をつなぐ線は、他の制御の移動を表わす線と区別するために、流れ線の折れ目の右側に ' $<$ ', ' $>$ ' の矢印ではなく ' $-$ ' を使い横方向に出入りがないことを示している。

(3) ブロック I F の構造を作る文を、 $$ と表わしブロック I F の構造の先頭は I F $$ と表わす。

FACOM OSIV/F4 AUTO FLOWCHARTER -831129- V-03 L-00 DATE 84-09-18 TIM

```

2<-----<#-<->      1      IF (A.NE.B) GO TO 50
| 1-----IF<B>      2      IF (A.EQ.B) THEN
| | 0>----->|      10
| | : | |      3 10      A=B
| | : | |      4      B=C
| | 0<-<#-<->      5      IF (A.NE.B) GO TO 10
| 1-----<B>      6      ELSE IF (A.NE.B) THEN
| | 0>----->|      20
| | : | |      7 20      A=B
| | : | |      8      B=C
| | 0<-<#-<->      9      IF (A.NE.B) GO TO 20
| 1-----<B>      10     ELSE IF (A.LT.B) THEN
| | 0>----->|      30
| | : | |      11 30     A=B
| | : | |      12     B=C
| | 0<-<#-<->      13     IF (A.NE.B) GO TO 30
| 1-----<B>      14     ELSE
| | 0>----->|      40
| | : | |      15 40     A=B
| | : | |      16     C=B
| | 0<-<#-<->      17     IF (A.NE.B) GO TO 40
| 1-----<B>      18     ENDIF
| | : | |      C
2>----->|      50
| 0---IF<B>      19 50   IF (A.EQ.B) THEN
| | : | |      20     A=B
| | : | |      21     B=C
| 0-----<B>      22     ELSE
| | : | |      23     A=B
| | : | |      24     C=B
| 0-----<B>      25     ENDIF
| | : | |      C
| 0---IF<B>      26     IF (A.EQ.B) THEN
| | : | |      27     A=B
| | : | |      28     B=C
| 0-----<B>      29     ENDIF
| | : | |      30     STOP
| | *S | |      31     END

```

図1. Fort flow77の出力例

3. コマンド

(1) 形式

FLOW77 データセット名 [SY (出力クラス | * | 出力データセット名)]

(2) オペランドの説明

データセット名 : Fortranソースデータセット名(標準形式)を指定する。

SY (出力クラス | * | 出力データセット名) : 結果の出力先を指定する。省略すると、システムはSPSTINGY [2] に処理を依頼して図1のような縮小リストをクローズドNLPに出力する。個人のデータセットに出力しておき、その後SPSTINGY [2] 等で縮小して出力する方法もある。

出力クラス : A, O, K, S, Uのいずれかを指定する。

* : 端末に出力する。

出力データセット名: 個人のデータセットに出力する。

4. 印刷行数の指定

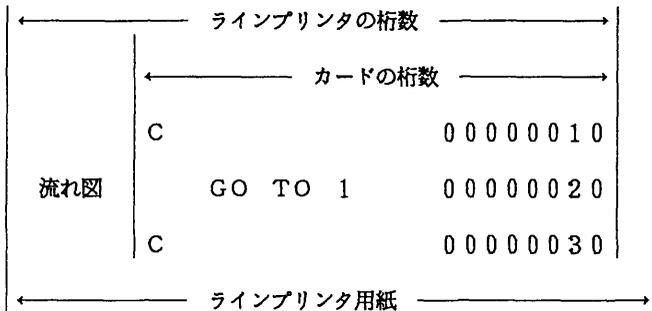
印刷するカードの桁数とラインプリンタの桁数を特別に指定することができる。ソースプログラムの先頭に

コラム1
↓

C=カードの桁数, LPの桁数

のような註釈行を挿入することによって行う。

ここで、印刷するカードの桁数とラインプリンタの桁数とは次の通り。



なお、指定しない場合の標準値は、カードの桁数が72、ラインプリンタの桁数が132、つまり、C=72、132である。

カードのシーケンシャルの部分が必要な時は、C=80、132とすればよい。

80欄のTSS端末へ出力する時は、たとえばC=50、80と指定するとよい。

指定された値は、再指定されるまで有効である。特定のプログラム単位以降を再指定するには、そのプログラム単位の直前のプログラム単位に属するEND行の直後に上の註釈行を挿入すればよい。

5. 使用上の注意

(1) データとして入力できるプログラムは、コンパイラにかけてエラーのないものが望ましい。システム自体は、文法のエラーチェックは行わないので、文法エラーがあると流れ線の正しさは保証できない。未定義文番号がある場合には、メッセージが出る。

(2) 一つのプログラム単位の行数は2000行以内に限る。

(3) 一つのプログラム単位内で、文番号の数500個以内、行き先文番号の数1000個以内に限る。

(4) 流れ線を書く部分は、(ラインプリンタの桁数-カードの桁数-16)桁しかないので、流れ線の多重度はその値を越える場合には、その分だけソースリストの右側の部分が失われる。

(5) Fortran 77で追加された“GOTO 変数名”の形の割り当て形GOTO文に対しては流れ線を引けないので“GOTO 変数名(文番号, 文番号, 文番号, . . .)”の形を使用すること。

(6) 標準形式Fortranプログラムでないと処理できない。

使用の効用として次のような点を挙げることができる。

(1) 特殊な入出力機器を必要としないので、簡単なコマンドで使用できる。

(2) プロッタ等で書いたものに比べ、流れ図としての品質は落ちるが、ソースプログラムがすぐ右側にある便宜により、品質の不足は補われる。

(3) ソース・テキストの変更を直ちに流れ図の変更に反映させることができる。

(4) プログラムの構造の具合が流れ線の複雑さとなって表れるので、プログラムのどこを工夫すればよいかという手掛かりを与える。

参考文献

1. 牛島, 高比良 流れ図つきソースプログラム作表システムFlowの使用について, 九大大型計算機センター広報, 11, 4, 284-288, 1978.

2. 吉田 データセット印刷プログラムSuper Stingyの使用について, 九大大型計算機センター広報, 16, 4, 440-450, 1983.