

## FORTRAN77プリティプリンタの使用について

藤村, 直美  
九州大学情報処理教育センター

河村, 豊実  
九州大学工学部情報工学科

<https://doi.org/10.15017/6768413>

---

出版情報 : 情報処理教育広報. 7 (1), pp.2-6, 1984-08. Educational Center For Information Processing, Kyushu University

バージョン :

権利関係 :



## FORTRAN77 プリティプリンタの使用について

藤村 直美\*, 河村 豊実\*\*

## 1. はじめに

情報処理教育センターで利用可能なOSIV/F4のFORTRAN77にはJIS規格どおりの固定形式とTSS向けの自由形式の2つのソース形式がある。自由形式は固定形式に比べて入力制限がゆるいため、TSSでの入力が容易であるためか、利用者が多い。しかしながら自由形式でプログラムを入力し終わった後、そのプログラムを見直そうとすると、大抵の利用者は段付け等を考慮せずに入力しているため、プログラムがゴチャゴチャして読みにくいことが多い。

ここで紹介するプリティプリンタはそのようなゴチャゴチャしたプログラムをもう少し見やすいプログラムへ変換するプログラムである。このプログラムを使用することによって、入力する時は自由形式の手軽さで入力し、あとで見直す時は人が見てわかりやすい形式にすることができる。わかりやすいプログラムはバグが入り込みにくく、またバグがあるとしても容易に発見できる。できるだけわかりやすいプログラムを作るように心がけて欲しい。

## 2. 使用例

ここでは自由形式で入力したFORTRAN77プログラムをプリティプリンタを用いて段付けする例を示す。

```

READY
E INPUT.FORT77 ..... 段付けする前のFORTRAN77プログラムを呼び出す。
                           すべての行が1桁目から始まっていて決してわかりやすいとは言えないプログラム。
E
L ..... 内容を確認する。
00010 *THIS PROGRAM COMPUTES FIRST N PRIME NUMBERS
00020 *
00030 INTEGER PRIME(5000)
00040 LOGICAL SW
00050 *
00060 READ(5,10)N
00070 10 FORMAT(I5)
00080 *
00090 IF(N.GT.5000.OR.N.LT.1) THEN
00100 WRITE(6,20)N
  
```

\* 情報処理教育センター

\*\*工学部情報工学科

```

00110 20 FORMAT(5X,11HINVALID N (,I6,1H))
00120 STOP
00130 ENDIF
00140 *
00150 PRIME(1)=2
00160 INDEX=1
00170 NUM=1
00180 *
00190 DO 500 NO=2,N
00200 200 NUM=NUM+2
00210 IF (PRIME(INDEX)**2.LE.NUM) INDEX=INDEX+1
00220 K=2
00230 SW=.TRUE.
00240 300 IF(SW.AND.(K.LT.INDEX)) THEN
00250 SW=(MOD(NUM,PRIME(K)).NE.0)
00260 K=K+1
00270 GOTO 300
00280 ENDIF
00290 *
00300 IF(SW) THEN
00310 PRIME(NO)=NUM
00320 ELSE
00330 GOTO 200
00340 ENDIF
00350 *
00360 500 CONTINUE
00370 *
00380 WRITE(6,30) (PRIME(I),I=1,N)
00390 30 FORMAT(5X,5I10)
00400 *
00410 STOP
00420 *
00430 END
KEQ52500I END OF DATA SET
E

```

RUN ..... 実行して見る。  
 FORTRAN 77 COMPILER ENTERED  
 END OF COMPILATION  
 00060 ?

00050 ..... データを入力する。

|     |     |     |     |     |
|-----|-----|-----|-----|-----|
| 2   | 3   | 5   | 7   | 11  |
| 13  | 17  | 19  | 23  | 29  |
| 31  | 37  | 41  | 43  | 47  |
| 53  | 59  | 61  | 67  | 71  |
| 73  | 79  | 83  | 89  | 97  |
| 101 | 103 | 107 | 109 | 113 |
| 127 | 131 | 137 | 139 | 149 |
| 151 | 157 | 163 | 167 | 173 |
| 179 | 181 | 191 | 193 | 197 |
| 199 | 211 | 223 | 227 | 229 |

END OF GO,SEVERITY CODE=00  
 E

END ..... エディタを終了する。  
 READY

RUN ..... 実行して確認する。

FORTRAN 77 COMPILER ENTERED

END OF COMPILATION

00060 ?

00050 ..... データを入力する。

|     |     |     |     |     |
|-----|-----|-----|-----|-----|
| 2   | 3   | 5   | 7   | 11  |
| 13  | 17  | 19  | 23  | 29  |
| 31  | 37  | 41  | 43  | 47  |
| 53  | 59  | 61  | 67  | 71  |
| 73  | 79  | 83  | 89  | 97  |
| 101 | 103 | 107 | 109 | 113 |
| 127 | 131 | 137 | 139 | 149 |
| 151 | 157 | 163 | 167 | 173 |
| 179 | 181 | 191 | 193 | 197 |
| 199 | 211 | 223 | 227 | 229 |

END OF GO, SEVERITY CODE=00

E

END ..... エディタを終了する。

READY

### 3. PRETTYコマンド

#### (1) 機能

自由形式 (VB) のFORTRAN 77のソースプログラムを段付けし、わかりやすい形式に変換する。

#### (2) コマンド形式

| コマンド   | オペランド                    |
|--------|--------------------------|
| Pretty | 入力データセット OUT (出力データセット名) |

#### オペランドの説明

① 入力データセット : 段付けしたい自由形式のFORTRAN 77のソースプログラムが入っている入力データセットを指定する。

② OUT (出力データセット) : 段付けした自由形式のFORTRAN 77のソースプログラムを出力する出力データセットを指定する。新しく順データセットを作成するので、実行前に同じ名前のデータセットがあってはならない。

#### (3) 使用例

```
READY
PRETTY INPUT.FORT77 OUT(OUTPUT.FORT77)
```

PRETTY INPUT.FORT77 OUT(OUTPUT.FORT77) .....

..... プリティプリンタを呼び出す。

INPUT.FORT77がわかりにくいプログラム。

OUTPUT.FORT77に段付けしたプログラムを出力する。

OUTPUT.FORT77というデータセットが新しく作成される。

READY

E OUTPUT.FORT77 ..... 段付けされたFORTRAN77プログラムを呼び出す。

E

L ..... 内容を確認する。

```
00010 *THIS PROGRAM COMPUTES FIRST N PRIME NUMBERS
00020 *
00030     INTEGER PRIME(5000)
00040     LOGICAL SW
00050 *
00060     READ(5,10)N
00070 10    FORMAT(I5)
00080 *
00090     IF(N.GT.5000.OR.N.LT.1) THEN
00100         WRITE(6,20)N
00110 20    FORMAT(5X,11HINVALID N (,I6,1H))
00120         STOP
00130     ENDIF
00140 *
00150     PRIME(1)=2
00160     INDEX=1
00170     NUM=1
00180 *
00190     DO 500 NO=2,N
00200 200    NUM=NUM+2
00210         IF (PRIME(INDEX)**2.LE.NUM) INDEX=INDEX+1
00220         K=2
00230         SW=.TRUE.
00240 300    IF(SW.AND.(K.LT.INDEX)) THEN
00250             SW=(MOD(NUM,PRIME(K)).NE.0)
00260             K=K+1
00270             GOTO 300
00280         ENDIF
00290 *
00300         IF(SW) THEN
00310             PRIME(NO)=NUM
00320         ELSE
00330             GOTO 200
00340         ENDIF
00350 *
00360 500    CONTINUE
00370 *
00380     WRITE(6,30) (PRIME(I),I=1,N)
00390 30    FORMAT(5X,5I10)
00400 *
00410     STOP
00420 *
00430     END
KEQ52500I END OF DATA SET
E
```

#### (4) 段付けの規則と制限

段付けの規則は次の通りである。

- ① 文番号は第1桁から5桁の間におく。
- ② IF文の深さ(レベル) 0の文は第7桁目から始め、レベルが1つ深くなるごとに3桁ずつ下げる。  
継続行はその文の始まる桁に対して5桁下げる。
- ③ ブロックIF文、ELSEIF文、ELSE文、ENDIF文は下げないで、IFブロック、ELSEIFブロック、ELSEブロックをそれぞれ3桁下げる。
- ④ DOループの範囲はDO文に対して3桁下げる。ただし、端末文は下げない。またDOループの範囲内のDO文がそのループと同一の端末文を持っている場合、そのDO文は下げない。

段付けの制限は次の通りである。

- ① 注釈行は段付けしない。
- ② 文の識別は開始行のみで行うため、段付けのキーとなる文(ブロックIF文などは)は1行に書くこと。
- ③ DO文は次の形式でなければならない。  
$$DO \Delta 1 \quad v = \dots\dots$$
  
ここで $\Delta$ は1個以上の空白とする。
- ④ IF文の深さが負になる文は段付けをしない。
- ⑤ IFの深さとDOループの入れ子の深さの合計が10の文まで段付けする。それを超える文は深さ10の文として扱う。

#### (5) 使用上の注意

- ① 自由形式のFORTRAN 77ソースプログラムしか処理しない。
- ② 出力データセットは新しく作成するので同じ名前のデータセットがあってはいけない。
- ③ 出力データセットは順データセットとして作成されるので、必要なら区分データセットへコピーすること。