システム開発の基礎教育について(その4)

1B-7

— COBOLで起こしやすいエラーとデバッグ手法 —

増田 秋俊, 今井 恒雄 (富士通) 荒木 雄豪, 平木しげ子, 中井 勝己,
中村 順 (京産大)

1. はじめに

興味と学習意欲を引き出し, 自信と達成感を持たせながら言語教育を進める上で, コンピュータの利用環境を整えることの重要性は明らかである。プログラム開発時にどのようなエラーを起こしやすく, どう調べたらよいかについて, また, 富士通のOSであるOSIV/F4 MSP の上でどのようなデバッグ環境を設定したかについて述べる。

2. COBOLで起こしやすいエラー

プログラム開発の段階で, 初心者は同じようなエラーを起こしやすい。エラーはコンパイル時のエラーと実行時のエラーに大きく分けられる。プログラム開発では, どこで, なぜエラーが起きたかが分かればよいわけだが, コンパイルエラーは, 行番号とエラー内容が表示されるので, エラーメッセージの意味さえ分かれば修正できる。

一方, 実行時エラーは, エラーメッセージだけではエラー原因がよく分からないケースが多い。我々は実行時エラーを中心に, 起こしやすいエラーをまとめ, 実習及び実務時の補助として使っている。ここでは, その中のいくつかの例を挙げる。エラーの中では, ファイルと表操作に関するエラーが最も多い。

(1) ファイル

- レコード長, ブロック数の指定ミス。
- ファイルのOPENのし忘れ。
- ファイルのAT END後のREAD。
- OPENしていないファイルのレコード領域にデータをセットする。
- CLOSEのし忘れ。

(2) 表操作

- 領域のクリア忘れ。
- 添字用の変数の初期化忘れ。
- 添字用の変数の桁数不足。
- 添字用の変数の値が範囲外。

(3) 演算

- 数字項目の値がニューメリックでないまま (例えば、ブランクが入っている) 演算する。
- 割り算の計算で, 分母がゼロになっているのに気がつかない。

(4) TSS特有のエラー

- COBOLでは73カラム以降はコメントとして扱われる。画面入力では, 73カラムの位置がわかりにくいので, 越えてしまう場合がある。

3. デバッグ手法

デバッグを効率よく行うには, TSSの有効利用と, メーカ提供のデバッグツールを適切に使うことと, その上でユーザ自身がデバッグの手段を臨機応変に講じられるように教育することである。

(1) TSSでの実習

教育では, プログラム編集から実行までを簡単な操作でできることが大切である。我々は, フルスクリーン型の端末を使いTSSの下で実習すること, プログラム編集, ファイル操作, プログラムの実行などを一連の処理として実行できることを可能にするため, メーカ提供ソフトのPFDを使ってCOBOL教育を行っている。

(2) デバッガ

OSIV/F4 MSP では, いろいろなデバッガが用意されているが, その中で非常に簡単で効果のあるのがCOBOLコンパイラの@OPTIONS機能である。これはCOBOLソースプログラムの先頭に, @OPTIONS文としてデバッグ機能を指定し, 実行させるものである。

この機能によって, どこで, どんなエラーが起きたかが分かる。

- STATEMENT

  エラーを起こした行番号を表示する。当たり前であるが, この機能なしでデバッグする大変さを考えるときわめて有効なものである。

Basic Instraction For System Development (No.4)

Akitosi MASUDA[1], Tuneo IMAI[1], Hugo ARAKI[2], Sigeko HIRAKI[2],
Katumi NAKAI[2], Zyun NAKAMURA[2]

1 Fujitsu Ltd. 2 Kyoto Sangyo Univ.

・CHECK

OCCURSで定義した領域の値と手続き部で指定した添字の値の範囲チェックをする。これも有効な機能である。

・FLOW

エラーを起こした行の直前に実行した行番号を，指定した行数分表示する。

これらの例を図－1に示す。

(3) ユーザがなすべきこと

これまで述べた環境で，デバッグを進めるわけだが，ユーザ自身が用意すべき手段として，次の二つがある。

a．DISPLAY命令

エラー原因をさがすために，自分のプログラムの中の調べるべき場所にDISPLAY命令を入れて状況をつかむ方法である。ありふれたことかもしれないが，これがデバッグの基本である（図－1参照）。

b．16進DUMP

データの内容が2バイト系の漢字コードあるいはバイナリの場合や，エラーのためどのような値が入っているか分からない場合には，DISPLAY命令では，その内容が分からない。

そのような場合には，16進表示をするしかない。そこで，我々は，16進DUMPルーチンをサブルーチンとして用意しており，プログラムの任意の箇所から任意の項目の値を16進表示を行えるようにしている（図－2参照）。

この二つをデバッガと組み合わせて，使いこなすことにより効率のよいデバッグができると考えている。

4．おわりに

COBOLの言語教育の実習を円滑に行うには，事前の環境設定とコンピュータの基本的な使い方，デバッグ環境の設定方法，エラー発生時の原因調査方法など簡潔にまとめた手引書を用意しておくことが，必要であると考える。

```
000010  @OPTIONS STATE,CHECK,FLOW
000020  IDENTIFICATION    DIVISION.
000030  PROGRAM-ID.       TEST88.
000040  DATA              DIVISION.
000050  WORKING-STORAGE   SECTION.
000060  01  CNT-AREA.
000070      02  CNT-1      OCCURS 9.
000080        03  CNT-2    OCCURS 6.
000090          04  CNT-3 OCCURS 3 PIC  9(2).
000100  01  I                       PIC  9(4).
000110  01  J                       PIC  9(4).
000120  01  K                       PIC  9(4).
000130  PROCEDURE         DIVISION.
000140  A10.
                  :
                  :
000220      ADD     1   TO  J.
000230      IF      J  >  6
000240              GO TO  A20.
000250      MOVE  ZERO  TO  K.
000260  A40.
000270      ADD     1   TO  K.
000280      IF      K  >  6
000290              GO TO  A30.
000300      MOVE   ZERO TO  CNT-3(I, J, K).
000310      DISPLAY "A40,I=" I ",J=" J ",K=" K
000320       ",CNT-3(I,J,K)=" CNT-3(I, J, K).
000330      GO TO  A40.
000340  A50.
000350      STOP RUN.
```

実行

ユーザによるDISPLAY情報（ラベル）
```
A40,I=0001,J=0001,K=0001,CNT-3(I,J,K)=00
A40,I=0001,J=0001,K=0002,CNT-3(I,J,K)=00
A40,I=0001,J=0001,K=0003,CNT-3(I,J,K)=00
```

CHECK情報（プログラム名）
```
JMP0820I-U :TOO    SUBSCRIPT/INDEX IS OUT OF RANGE. PGM=TEST88. LINE=300.1. OPD
=CNT-3 (3)
```
（項目名　次元数）

STATEMENT情報
```
STATEMENT INFORMATION
   PROGRAM-NAME     STATEMENT NUMBER          COMPILATION TIME
   ---------------- ------------------        ----------------
   TEST88 プログラム名          300.1          88-12-07 15:27:56
```
（行番号（小数点の値はその行の動詞番号））

FLOW情報
```
FLOW INFORMATION
   TEST88 プログラム名
                140      160    200.1      210    250.1      260    300.1      260    300.1
    260    300.1      260    300.1
```

図－1．デバッグ情報

```
----------------------- (TRSUB2) HEX-DUMP  LGT=00054 -----------------------
F9F0F0F140F140F140F0F140F240C9E2C9D2C1E6C140C8C9C4C5D4C9404040404040400000000
9 0 0 1   1   1   0 1   2   I S I K A W A   H I D E M I
----------------------- (TRSUB2) HEX-DUMP  LGT=00054 -----------------------
F9F0F0F240F240F240F0F240F140C9E3E4D2C940C8C9D9D6E2C9404040404040404000000
9 0 0 2   2   2   0 2   1   I T U K I   H I R O S I
```

図－2．16進DUMP