

システム開発の基礎教育について(その3)

1B-6

— COBOL 言語教育と利用環境 —

今井 恒雄 荻田 直史 増田 秋俊(富士通) 荒木 雄豪 平木しげ子(京産大)
古谷 千恵(東大生研)

1. はじめに

我々はこれまでシステム開発の基礎教育について実践を通して、そのあるべき姿を研究してきたが^(*)1)、^(*)2)、今回は、実社会の事務処理の分野で最も広く使われているCOBOL言語について、その教育の在り方を述べる。

2. COBOL 言語教育のポイント

COBOLとFORTRANの言語教育を比較すると色々な面で違いがあるが、その一つに、FORTRANは教育する教師自身が自らの研究でも使っているケースが多いのに対して、COBOLは研究で使うケースがあまりないという事実がある。

これは言語教育に大きな影響をもつと考えられる。実社会の事務処理の分野では現在COBOLが主流になっている。従ってCOBOLを使ったシステム開発のノウハウは、システム開発に実際に携わっている人や組織に蓄積されている。しかし、学校教育の中では、教師自身が実際にシステム開発を行う機会が少ないことから、こういったノウハウの蓄積も少ないのが実態であろう。

我々は幸いにも大学の事務処理の業務開発に実際に関わり開発を担当してきた。また言語教育にも携わってきているので、その経験を踏まえてシステム開発に直接結びつくCOBOL言語教育の在り方について、そのポイントをまとめてみた。

(1) まず基本命令から

ともすれば時間の制約などからCOBOLの文法教育が中心になりがちであるが、まずどの機械でも使えるように、COBOLの基本的な命令を徹底的に教えることに絞ることが重要である。

COBOLの命令の中には、うまく使えばステップ数が短くなるとか、プログラムが見易くなるといった機能も多くあり、学生もそういった機能にはより興味を持ち、知っていることに優越感を持ちたがるものである。しかしそうした機能は基礎教育が終わった後、自分で学べばよいことである。

(2) 理解しにくいこと、間違いやすいことを重点に

COBOL言語の仕様そのものと、使用するコンピュータの仕組みの両面に、初心者にとって理解しにくいことや間違いやすいことがある。

基本命令の教育にあたっては、こうしたことを充分加味して教育すれば、文科系の学生にとっても分かり易いものになるだろう。

ここではその例をいくつか挙げておく。

a. ファイル(理解しにくいことの例)

初心者にとってファイルの概念は、一般的に言って分かりにくいものの代表である。

①ファイル編成

順次ファイル、区分ファイル、索引順次ファイルなど

②ファイル操作

ファイルの作成、修正、複写、内容表示など

③プログラムとの対応付け

COBOLプログラムの中でのファイルの定義と、実際のファイルとの対応付け

④ファイル処理

COBOLプログラムの中でのファイルのOPEN、CLOSE、READ、WRITE処理

これら4つの概念を自然に分かり易く理解させることは容易ではないが、非常に重要なことである。

b. COBOLでの表操作(間違いやすいことの例)

OCCURSを使ったn次元の表操作は間違いやすい機能の代表である。

①データ部と手続き部の整合性

表の大きさと添字に使う項目の桁数

②添字の初期化

表の中での位置を表す添字の初期化

③範囲外の指定

OCCURSで定義した大きさを超えた添字指定による処理異常

これら表操作は、理屈では理解できても、処理上の間違いを起こしやすい機能である。

(3) 実習をできるだけ多く

「習うより慣れる」と言うことはどのような場合でも言われるが、COBOLの場合も同じであり、実習にできるだけ多くの時間を割くこと、適切なテキストを用意することが大切である。我々はシステム開発を前提として

- ・ファイルを使わない入出力
DISPLAY, ACCEPT命令を使う
- ・簡単なファイル処理
- ・ファイル処理と帳票出力
- ・表操作

など、易しい機能から少しずつ新しい機能を加えていき、コンピュータに慣れながら、システム開発を意識したCOBOLを、自然に体で修得できるようなテキストを作成している。

3. コンピュータの利用環境

言語教育をスムーズに、しかも達成感を味あわせながら進めるには、使い易いコンピュータの利用環境を整えることが重要である。具体的には

- ・TSS中心の実習法
- ・デバッグの方法
- ・起こしやすいエラーとその現象

などを整理し、テキストとして用意しておくのがよい。これまでの経験上、初心者はだいたい同じような間違いを起こすものである。コンピュータの場合、あらかじめ基本的な使い方を教育すれば、テキスト通りならば一応使えるが、デバッグの段階でコンパイルエラー、実行時エラーなどのメッセージが出た場合、メッセージの意味は初心者には非常に理解しづらい。ましてメッセージが英語だとなおさらである。

その原因はメッセージが初心者向きになっていないからである。実行時にユーザプログラムが異常終了したなどというメッセージがでて、デバッグの方法を知らないと、端末の前でじっと考え込むか、教師に教えてもらおうしかない。このような環境のためにコンピュータは難しいものだと思ひ嫌気がさし、脱落していくケースがあることを考えると、COBOLの言語教育の環境を整えることはきわめて重要である。

(1) TSS中心の実習

プログラム開発の開発環境の基本はTSSである。プログラムの作成、修正、実行までをTSSの下で簡単な操作でできること、帳票出力などのバッチ処理にも簡単に使えることが大切である。

(2) デバッグの方法

プログラムのデバッグは、利用環境の違いで大きく能率が変わってくる。

- ①メッセージが日本語であること。
- ②端末にも、バッチの出力リストにも、日本語で出ること。
- ③間違った行がわかること。
- ④メッセージの意味がわかりやすいこと。

など、できるだけ初心者にはデバッグのやりやすい環境を用意することが大切である。

(3) 起こしやすいエラーとその現象

初心者がどういうエラーを起こしやすいかは経験的に把握できる。そのような事例をテキストに載せておくことはデバッグの能率を上げるのに大いに役立つ。

4. おわりに

システム開発を前提としたCOBOL言語教育の在り方について述べたが、デバッグ環境の整備の中で、ハードウェア、ソフトウェアを提供するメーカーの責任は重大である。つまり細かいことや基本的なことの積み重ねではあるが、それらの中にソフトウェアの生産性向上に直接関係する問題が多々あることを忘れてはならない。

参考文献

- *1 今井他 システム開発の基礎教育について
情報処理学会 第36回全国大会 6K-7 (1988)
- *2 今井他 システム開発の基礎教育について(その2)
情報処理学会 第37回全国大会 4S-8 (1988)