

誰にでも出来ないOA化から誰にでも出来るOA化へ（その4）

1M-1

—誰にでも出来るための訓練法—

今井 恒雄、増田 秋俊（富士通） 平木 しげ子（東京家政学院筑波短大）

荒木 雄豪（京産大） 古谷 千恵（東大生研）

1. はじめに

これまで、「誰にでも出来るOA化」のテーマのもとに、全くの素人が行うOA化の具体的な方法についてシリーズで発表してきたが、今回はそのような人達に対するCOBOL言語教育では、いつ、どのような内容のプログラムを作らせたらいのか、何故その時点で必要なのか等について、我々が作成・使用している訓練書に基づいての分析を発表する。

2. 訓練書とそこに含まれる問題数

「繰り返し」によってプログラムの作成と、システム設計を覚えることを原則としているため、訓練書に含まれる問題は多い。自習書として使用されることを考慮し、すべての問題には原則として解答例がついている。

	(問題数)
NO.1: COBOL による業務プログラムの作成 I (平木著)	92
NO.2: COBOL による業務プログラムの作成 II (平木著)	79
NO.3: ファイル処理入門 (平木他著)	187
NO.4: システム設計入門 (平木他著)	107
NO.5: PFD 入門 (今井他著)	39

(註: 上記図書の出版社はすべて「恒星社恒星閣」)

3. NO.1の訓練書に含まれるプログラムの内容分析

上記のNO.1 (プログラム問題集) 及びNO.3 (文法、基本システム設計問題集) の2冊が初級プログラマ養成のための訓練書となっている。ここでは、NO.1のプログラム問題集には、どのようなプログラムがどのような順番で入っているのか、どのような量をこなしていくのか等について説明する。なお、ここでの進度は短大で行っている速度、週1回の授業で約3ヵ月かかって終了するものとする。

【第1段階】 約3週間 (授業数3回) で25本のプログラムを作成する。

内容: DISPLAY 命令を使って自分の氏名やイニシャルを表示したりするごく簡単に短いプログラムを見本のプログラムを見ながら作ること、及び、ごく簡単な変更処理。

目的: コンピュータの操作 (含むキーボード操作) に馴れさせる。COBOL プログラムのイメージをつかませる。プログラムの作成と実行との違いを理解させる。コンパイル・エラーとは何か、及び、

A training method for system development that anyone can do (No.4)

Tuneo IMAI¹, Akitosi MASUDA¹, Sigeko HIRAKI², Hugo ARAKI³, Tie HURUTANI⁴

1 Fujitsu Ltd. 2 Tokyo Kasei Gakuin Tukuba Junior College 3 Kyoto Sangyo Univ.

4 Institute of Industrial Science, University of Tokyo.

そこで出たエラーの直し方を覚えさせる。特に、ピリオドをつけ忘れてたり、或いは、つけてはいけない箇所にピリオドをつけるとコンパイル・エラーになる（ことがある）等を通し、コンピュータでは1字でも間違っただけはいけないことを身体で覚えさせる。

【第2段階】約3週間（授業数3回）で13本のプログラムを作成する。

内容：ACCEPT命令、DISPLAY命令を使いコンピュータとのやりとりを行うプログラム（例えば、簡単なゲームプログラム）を見本のプログラムを見ながら作ること、及び、ごく簡単な変更処理。

目的：「コンピュータへデータを入力する」「コンピュータとやりとりをする」とはどういうことかを理解させる。プログラム中には条件文が入っているので、ピリオドのつけ方を間違えると目的と違う結果が出ることを通して「注意深くプログラムを作ることの必要性」を身体で覚えさせる。

【第3段階】約1週間（授業数1回）で5本のプログラムを作成する。

内容：これまでに作成したプログラム（ACCEPT命令、DISPLAY命令使用）をファイル処理（OPEN, READ, WRITE, CLOSE命令使用）に変えてある見本のプログラムを見ながら作ること、及び、応用問題の作成。

目的：同じ結果の出るプログラムでも、ファイル処理では長いプログラムになること、汎用機では、プログラムを実行する前に実ファイルへの割当（ALLOCATION）が必要なことを覚えさせる。この割当処理を忘れてたり間違ったりすると実行時のエラーとなるので、コンパイル・エラー以外にも、実行時のエラーもあること、この修正はコンパイル・エラーより遙かに難しいことを理解させる。

【第4段階】約5週間（授業数5回）で49本のプログラムを作成する。

内容：1つの入力ファイル（例：学生マスターファイル）から帳票を出力するプログラム（ソートを含む）を作る。この段階では、部分的な解答のみが載っている、或いは解答は省略してある。

目的：ファイル処理では、レコード長が1桁違っても、ブロック数を間違えても、OPEN命令を間違えても実行時のエラーが出ることを理解させる。（これは、いろいろなファイルを使用し、数多く作らせないと身につかない。）この時点で、チャンスがあれば実際のシステム開発に、帳票出力の担当者として参画させるのが、やる気を起こさせる最も効果のある方法である。

5. おわりに

数年の経験を持つベテランプログラマでも、頭を抱えて数時間或いは数日間もエラーの修正が出来ない原因に次のようなことがある。

- ① ピリオドのつけ忘れ、或いはつけてはいけない箇所につけてしまう。（論理ミス）
- ② ファイルのオープン指定のミス。（入力を出力としてしまう等）
- ③ 演算項目のクリアのし忘れ。
- ④ テーブル（OCCURS）位置の指定ミス。

等があるが、この内の①は第1段階、第2段階のプログラムの作成により、②③は第3段階、第4段階のプログラムの作成により、初期の時期に徹底的にエラーを起こさせないような習慣を身につけさせることが出来ると考えている。初期の時期なら徹底出来ることも、時期を逸すると身につかないことが多いので、「教える時期を選ぶ」のも教育では大切なことではなかろうか。

なお、第2段階のプログラムの作成に入ると、（自分から）文法に興味を持ち出すので、教育効果を考え、この時期まで文法は一切教えないことにしている。