

る)に要する時間を節減したいという希望から生まれたものである。

従って市販されている参考書とはちがいで、えらばれた問題ひとつを最後まで活用していくのが第1の特長である。こうすれば増加分等はカードの挿入ですむからである。

第2の特長はステートメント等の紹介は、プログラマーの要求が流れ図に示されたときに取り上げられるということである。こうすれば、流れ図とステートメントとの対応がよくつかめるからである。

しかしながら欠点も大きい。

まず、最後まで活用しようとしているひとつの問題としてどんなものを選ぶかということである。

欲をいえば三重 Dō 文が使える位まで活用したいものである。

一例をあげると

1.  $(a+b+c)/2$  から始めるが実は「ヘロンの公式」

2. 二次方程式の根

などを手がけてみた。

実際はフローがかわるごとに、別に類題を与えているのである。ステップ方式に採用された基題はステップごとの変身ぶりに利用していきだけで、本来はそこでのステートメントの理解に主眼があることには違いないのである。

つぎに困ったのは受講者のノートと黒板とのかねあいである。

指導者は黒板に流れ線とか、挿入すべきプログラムを記入すればよいが、文番号がかわったり、黒板が複雑になってくると、とてもノートに同じように写し書きができなくなる。

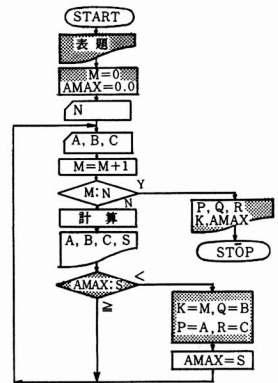
どうしてもその都度フローとプログラムをノートし

なければならない。

そのために下表のようなものを工夫してみたわけである。

この用紙を13枚として手渡ししておく、紹介ステートメントごとに赤でぬりつぶすようにしてみた。しかしどうも浮き上らない不鮮明なうらみが残ってしまう。

そこで、つぎのような用紙を配ることにした。



そして、変化した分のみのプログラムと新規のフローを記入させるようにしてみた。

でも、これではフローの変化はよく見えるけれど、プログラムがすぎだらけでしまらない。

結論としては、ステップ数だけスライドを作り、同時にステップ数だけの印刷物を手渡ししておくことになった。スライドを写していたのでは意味がないので、印刷物を配らなければならないのである。

ときに県内でもミニコン導入が語られるようになった。ここに機器選定のひとつの手がかりを述べてみたい。

(1) あるミニコンは、ソースプログラムを読み込ませる前後でシステム・プログラムを読み込ませなければならない。ただ、ソースプロが変るたびにシステムを入れなければならないので面倒なのである。

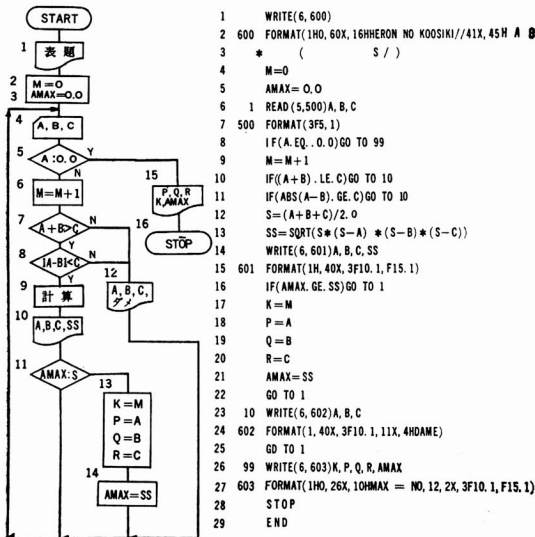
ところが、別のミニコンは一度システムを読み込んでおけば、あとは、それぞれのソースプロだけでよいことになっている。実習時間の制約からみて問題があろう。

(2) 「デバック」は知られていることだが、「アンバック」も必要である。あえて要所にエラーを組み込んでおき、そのミニコンがどの程度のエラーメッセージを示すかを検定するのである。エラー指示が細かく親切なのはよいが、容量にいくこまれたのではユーザープロの利用領域が少なくなる。

この「アンバック」作成は、よほど吟味しておかないとコンピュータ実習の効果を乏しいものにする。

この「アンバック」の利用は、機種選定のひとつの基準とも考えられるので、慎重に取り組むべき問題であろう。

これはまた、やがてプログラムを評価する上でも重要な資料を提供してくれるものと思われる。



```
1 WRITE(6, 600)
2 600 FORMAT(1H0, 60X, 16HHERON NO KOOSIKI//41X, 45H A B
3 * ( S / )
4 M=0
5 AMAX=0.0
6 1 READ(5,500)A, B, C
7 500 FORMAT(3F5, 1)
8 IF(A.EQ..0.0)GO TO 99
9 M=M+1
10 IF((A+B).LE.C)GO TO 10
11 IF(ABS(A-B).GE.C)GO TO 10
12 S=(A+B+C)/2.0
13 SS=SQR(S*(S-A)*(S-B)*(S-C))
14 WRITE(6, 601)A, B, C, SS
15 601 FORMAT(1H, 40X, 3F10. 1, F15. 1)
16 IF(AMAX.GE. SS)GO TO 1
17 K=M
18 P=A
19 Q=B
20 R=C
21 AMAX=SS
22 GO TO 1
23 10 WRITE(6, 602)A, B, C
24 602 FORMAT(1. 40X, 3F10. 1, 11X, 4HDAME)
25 GD TO 1
26 99 WRITE(6, 603)K, P, Q, R, AMAX
27 603 FORMAT(1H0, 26X, 10HMAX = NO, 12, 2X, 3F10. 1, F15. 1)
28 STOP
29 END
```