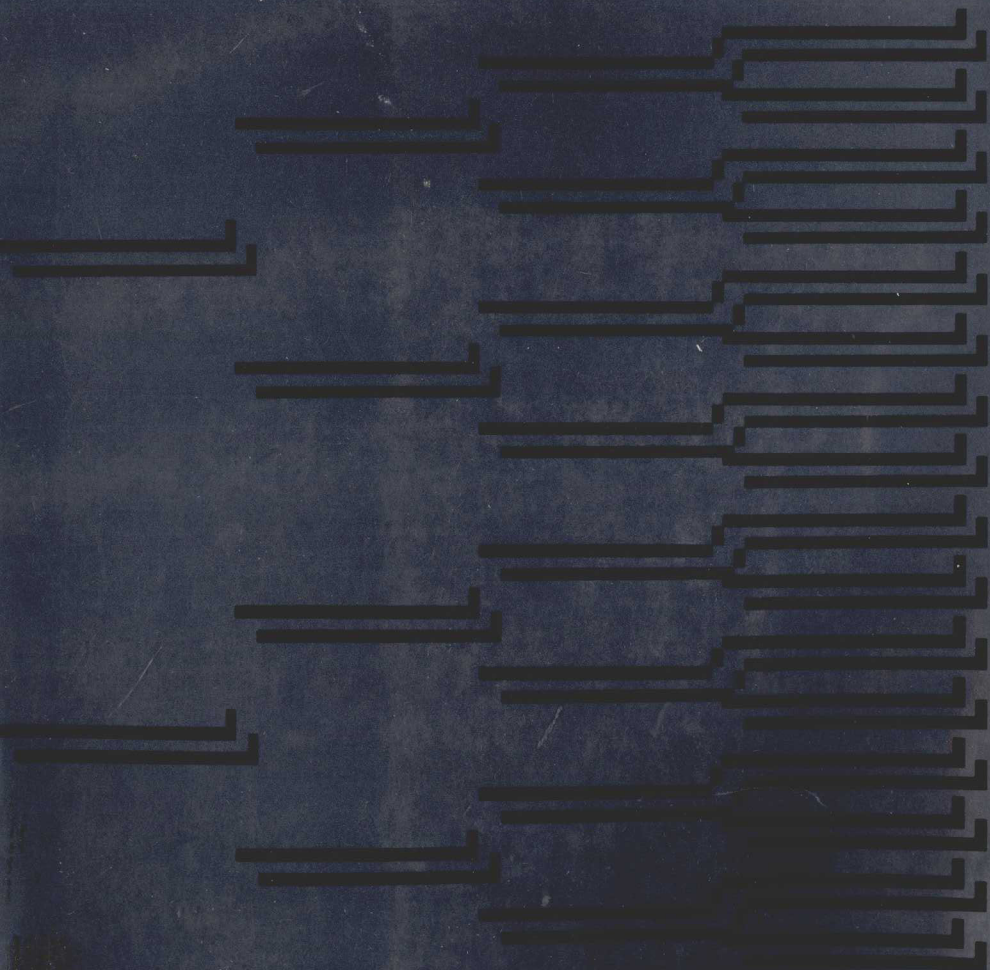


FORTRAN

プログラミング

二宮理恵 著



好学社

FORTRAN

プログラミング

二宮理憲著

マグロウヒル好学社

著者について

にの みや まと き
二 宮 理 意

1962年 東京都立大学大学院理学研究科修了
文部省統計数理研究所を経て

現 在 青山学院大学理工学部教授，医学
博士

専 攻 生体情報工学，医学統計学
1974年，カルガリー大学客員教授
1975年，ブリティッシュ・コロンビア
大学客員教授

主な著書

【電子計算機——基礎と応用】 新光閣，1967年

【PL/Iプログラミング入門】 森北出版，1977年

FORTRANプログラミング

定価1,800円

昭和54年4月20日 初版発行 ©

〔無検印制〕 著 者 二 宮 理 意

発行者 川 口 央

発 行 所 (株) マ グ ロ ウ ヒ ル 好 学 社

東京都中央区銀座4-14-11 (七十七ビル)

〒104 京橋局私書箱281 電話03-542-8821

無断転載複製を禁ず

広済堂印刷(株) 印刷・製本

Copyright © 1979 by McGraw-Hill Kogakusha, Ltd. All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic mechanical, photocopying, recording, or otherwise, without the prior written permission of the publisher.

まえがき

昔の寺小屋では、ヨミ、カキ、ソロバンを教えていたが、これは情報処理活動をする上での最も基本的な訓練の1つといえよう。この情報処理活動での優越性こそが人間を人間たらしめる——人間と他の動物とを区別する——特徴といえる。

現在では情報処理のための道具としては電子計算機があり、これを上手に使用えば、人間としての生活をより豊かにすることができるであろう。現在の社会で生活するにおいて、少なくとも電子計算機とは何であるかを理解しておきたいものである。そのためにも、電子計算機を少し使ってみたいものである。

この本では、電子計算機の基礎的な使い方について述べている。電子計算機を使うには——何か仕事をさせるには——、その仕事の中味を計算機の方でもわかるようなコトバ——プログラミング言語——で書かなければならない。その中の1つにFORTRANがある。これはいろいろなところで広く使われており、使用するに便利なプログラミング用語である。

本書は、著者の20年に近い計算機についての教育経験から生み出された方法によるもので、例題を中心に述べてあり、プログラミング言語の特徴がとらえやすくなっている。したがって、他のプログラミング言語PL/I, COBOL, BASICなどを学ぶ場合の足がかりともなるであろう。

この本では、JIS FORTRANの7000水準をもとにしているが、教育的立場からその範囲外の機能も一部取り入れてあり、また、範囲内のものでも省略したものがある。

最後に、原稿の整理と校正に並々ならぬ御援助をいただいた川原田稲子、さし絵を描いていただいた佐藤道子の両氏、およびいろいろと御協力いただいた青山学院大学情報科学研究センターの諸氏に謝意を表したい。

1979年2月

著 者

目 次

まえがき

0	電子計算機と言語	1
0-1	電子計算機と人間	1
0-2	フローチャート	4
0-3	言語の形式	11
0-4	計算機の言語の形式	14
1	簡単なプログラム	19
2	四則演算	25
3	実数型と整数型 (I)	29
4	実数型と整数型 (II)	32
5	絵を描くには	36
6	くりかえしの計算	40
7	数列の和	48
8	DOステイトメント (I)	52
9	DOステイトメント (II)	57
10	DOステイトメント (III)	60
11	データの読み込み (I)	64
12	データの読み込み (II)	70
13	平均の計算 (I)	73
14	平均の計算 (II)	77
15	添字付き変数 (I)	82
16	添字付き変数 (II)	86

17	グラフを書くには	89
18	数値計算	100
18-1	方程式の解	100
18-2	面積を求める	102
19	分類について	109
20	文字処理	116
21	FORTRAN 文法	126
21-1	代入文	126
21-2	制御文	129
21-3	入出力に関する文	134
21-4	DATA 文	144
21-5	宣言文	144
21-6	文関数定義文	150
21-7	副プログラム	150
21-8	END 行	154
FORTRAN 問題		157
練習問題解答		173
FORTRAN 問題解答		198
索 引		257

0 電子計算機と言語

0-1 電子計算機と人間

電子計算機という言葉はもはや子供向けの漫画にも登場してくるようになっているので、ここでくどくどと解説する必要はないものと思われる。しかしそのような場合それは単に権威づけのためにのみ用いられている場合が多いので、やはり少しばかりの説明は必要であろう。

電子計算機は一種の機械である。機械というものはわれわれのいろいろな仕事をする上で手助けとなるものである。ところで電子計算機という機械の仕事の分野は、他の一般的な機械といちじるしく異なっており、さらにその使用法も他の機械と比較するのは困難なほど広範囲である。

電子計算機は名前の通り計算する電子的な装置をもつ機械として作られたが、今日では、「計算」をかなり広い意味で解釈してもしきれないほどにいろいろな分野で用いられている。

ここで述べようとしている電子計算機と、一般にいわれている電子卓上計算機（電卓）とははっきりと区別しなければならない。もちろん最近の電子計算機は机の上に置けるほどの小ささでありながらかなりの性能をもっているものもあるが、いわゆる電卓というものは四則演算と少しばかりの関数の計算（たとえば $\sin$, $\cos$, $\log$ など）ができるのが普通である。

これに対して電子計算機とは、このような電卓とそれを操作する計算係の人間をいっしょにしたようなものである。いい換えると電卓は加算とか割算とか1つずつしかできないが、操作係は上司の命令に従ってこの1つずつの計算を積み重ねて結果をだすのである。これに対し電子計算機では1つの計算がすむとすぐ次の計算が自動的に行なわれるようになっている。そうして次にどんな計算をすべきかは、電卓の場合はその操作係が自分の頭で記憶しているが、電子計算機ではその記憶装置の中に記憶されており、その記憶されている命令に従って1つずつ計算を行ない結果をだすのである。このような一連の命令をプログラムという。電子計算機はこのプログラムなしでは何の働きもしないし、またプログラムをいろいろと変えればそれに従っていろいろな仕事をやってくれる。

電子計算機とは電卓とそれを操作する人間をいっしょにしたようなものであるが、基本的には図0-2のような構造をしている。計算係は上司の命令による



図 0-1 電卓とそれを使っている人間

電子計算機とはこのように電卓を使っている人間と電卓をいっしょにしたようなものである。

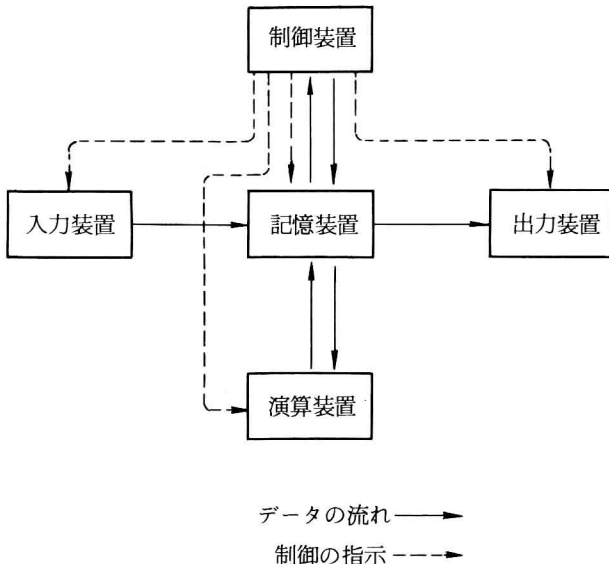


図0-2 電子計算機

計算指示表を目で読みながら、その指示通りにデータをもとにして計算している。計算の中間結果を頭に記憶する場合もあるが、あまり多い場合はメモ用紙に記入するであろう。

電子計算機においてはプログラム（計算指示表に対応）とデータは入力装置から入ってくる。それらはとにかく**記憶装置**（頭またはメモ用紙に対応）にたくわえられ、演算装置（電卓に対応）でプログラムに従っていろいろな計算がなされる（これはちょうど計算指示に従って電卓でいろいろな計算をしているのに対応）。計算の結果は出力装置により印刷される（計算係が手で伝票を書くのに対応）。電卓では数値は電卓にたくわえられている場合もあるが、その処理手順は人間の頭ないしはメモ用紙にある。これに対して電子計算機では数値も手順も機械の内にあってある。この点が重要な違いである。計算係にいろいろな計算をさせたければ、計算指示表とデータをいろいろ変えてやればよい。計算機でもプログラムとそのデータを変えれば、それに対応するいろいろな計算ができる。それ

では計算機ではどんな計算が可能かという問題が起こるであろうが、しかし計算機にできることは理論的な判断だけである。あるいはいい換えると四則演算（ $+$ 、 $-$ 、 $\times$ 、 $\div$ ）と条件の判断といってもよいであろう。しかもそれらを有限回使うだけである。一般論としてはこのような表現でよいであろうが、実際問題としては、計算機を 100 時間以上も用いなければ結果がでないような計算は、計算できないとはいえないにしても非常に困難な計算といえよう。また計算するために必要な記憶する場所が、現在作られている大型計算機の主記憶装置の数倍をも必要とするような計算もやはり困難な計算であるといわねばならない。計算機の計算速度も速くなり、記憶装置も年々大きくなっているのに 10 年前では困難であった計算が今では計算可能なものになっているのは事実である。このような傾向は今後とも続くであろう。計算機に命令するための言語——計算機言語——にはいろいろなものがあるが、この本では FORTRAN (FORmula TRANslatan の略) について述べる。

0-2 フローチャート

われわれが 1 つのある問題の解析をしようとする場合、ただばくぜんとその問題をながめていたのではいつまでたっても解決しないであろう。少なくとも、この場合はこうなるとか、ああなる、といった分析をするであろう。これを頭の中でしていたのではまともにくいので、図に書いて、一方で分解されたものと全体との関係も、はっきりと見なければならぬであろう。このような手法をフローチャートを書くといい、計算機での仕事の 1 つの基礎となっている。

1 つの例として、電子計算機と直接の関係はないが、信号機のついている交差点での渡り方（始めにいた地点より対称な地点に行く方法）についての道すじについて述べてみよう。交差点と一般的に述べたのではいろいろな交差点が考えられるから、新宿の伊勢丹の角の交差点に限定しよう。しかし伊勢丹のような大きな建物には、いろいろな角があり、そのそれぞれの角には、交差点があるかもしれない。

私は最近あの付近をよく見ていないので変わっているかもしれない。私が今問題にしようとしているのは明治通りと、四谷方面から来て新宿駅の東口へぶつかる道路の交差点のことである。

どうもくどくどと述べてしまったが、問題の始まりは交差点の渡り方であった。しかし、上記の（ ）内のような表現では少しわかりにくい点があると思われたので、具体的な例として伊勢丹の角を考えてみたのである。ところが図0-3の伊勢丹の地点と対称の地点に何も書いていないことからしてわかると思うが、そこになにがあったか私は知らないのである。もちろん、このようなことは問題と本質的なつながりはない。問題の本質は図0-4にあるように交差点の角をA, B, C, Dと名づけておけばよいのである。そうして、これにおいてAからC, (あるいはCからA), BからD, (あるいはDからB) へ行くにはどのようにしたらよいかを考えればよいのである。抽象的に (始めにいた地点より対称な地点に行く方法) といっていたのでは——もちろんこの問題は簡単なものだから（ ）内の表現でも十分に理解されようが——理解されない場合が多い。しかし実際の例を持ちだして、その例に引きずり回されて、その問題の本質を見失ってはならない。具体的な実際の例にひそんでいところの問題の本質をさぐりだし、図0-4のような抽象的な、そのような問題であればどんな場合にでも成立するところの規則を発見しなければならない。

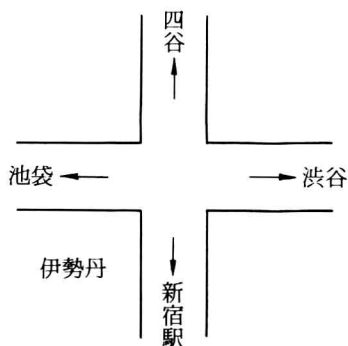


図0-3 交差点

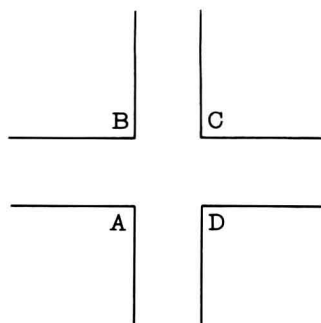


図0-4 交差点

いまA地点にいるとしよう。①Bの所の信号機を見る。青だったらBへ進む。そうしてそこで、次にCの信号機を見てそれが青になったらCへ進む。これでCへ到着することになる。①で、Bの所の信号機を見る。青だったら……といっておいたのだが、それが青でなく、黄色だったらどうなるのだろう。そのと

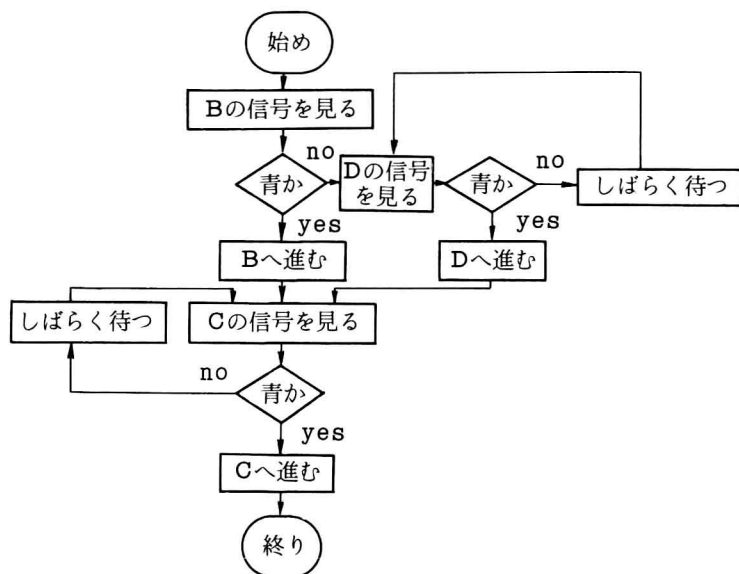


図0-5 フローチャート

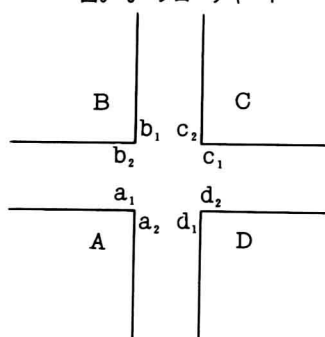


図0-6 交差点の信号機

(A, B, C, Dは角の名称, a_1 , d_2 などは信号機の名称)

きはしばらく待てばDの信号が青になるはずだから、しばらく待ってからDへ進み、次にCの信号が青になったらCへ進む。①でもしBの所の信号が赤だったら、Dは青になっているはずだからDへ進み、次にCの信号が青になったらCへ進む。

いま読んだ文章を見ていると、なんと面倒なことをくどくどと書いているのだらうという気がすると思う。だれもがこのような文章を好んで読もうとはしないであろうし、書く方でも気が進まないのである。

このようなときわれわれは文章ばかりでなく図を用いて、このようなことがらを明確にしようというのである。

少し考えてみれば図0-5のようなものができるであろう。もう気がついたと思うが1つの仕事は□の中に書き、判断は◇の中に書いてある。これが普通の方法である。

交差点での渡り方について考えてきたのだが、始めは具体的な問題であったのを図0-4のように抽象的な（それほどでもないが）問題にして、その問題における解答を1つの文章に書いてみた。しかし、文章ではわかりにくいためフ

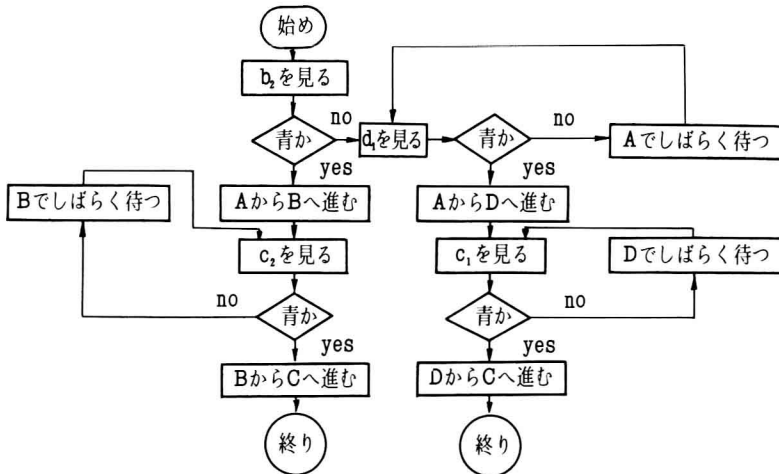


図0-7 フローチャート

ローチャートに書けば、それは図0-5のように簡単になる。だが、実は問題のとらえ方が明確でなかったので、もう一度、図0-6のように交差点とか信号機をはっきりと定義して考え直したのである。図0-7は、そのようにしてできたフローチャートであるが、これでは複雑すぎるので図0-8のように変更したのである。

この一連の作業をみていると、フローチャートを書くことによって問題のあいまいさが発見されたり、あるいは仕事のむだが発見されたりしている。こ

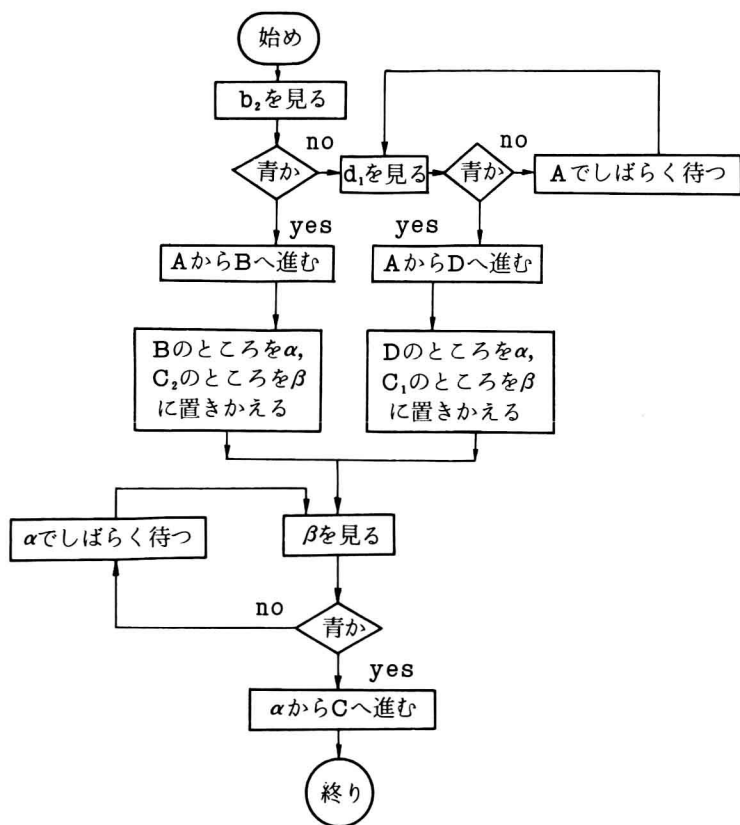


図0-8 フローチャート

ではなかったが、フローチャートを書くことによって、誤りを発見できることもあるのである。要するにフローチャートを書くことは、問題の解析のための手段として有効に働くことがわかるであろう。

図0-8と図0-9にある□内のBのところを α 、 c_2 のところを β 、などという仕事の内容は始めての人にはわかりにくいかもしれないが、電子計算機で実際の仕事をするためには、このような仕事は非常に重要なものである。

また図0-8と図0-9とでは□を入れた場所が異なるだけであるが、図0-9の方が図0-8より簡単になっていることに気がつくであろう。フロー

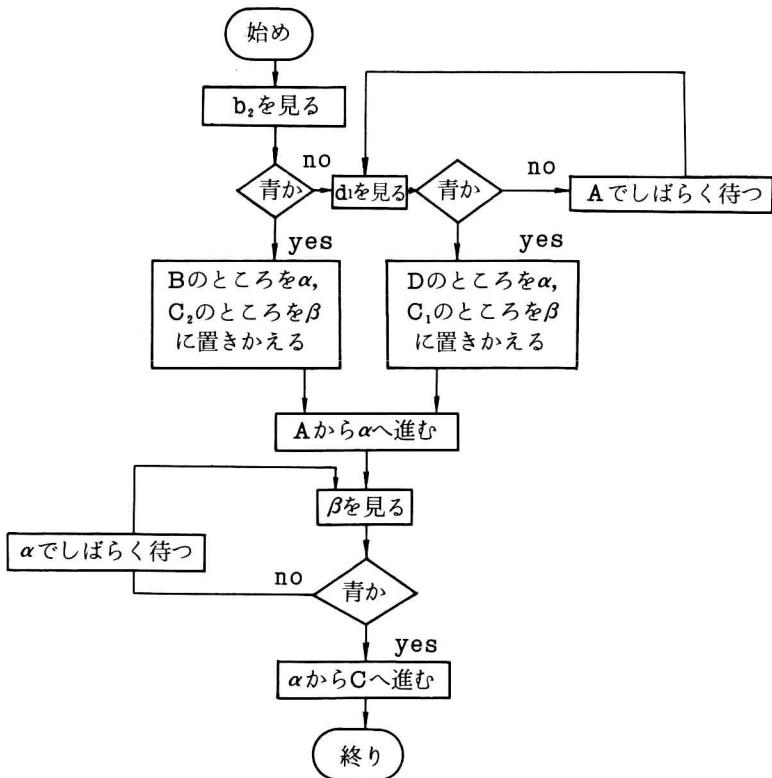


図0-9 フローチャート

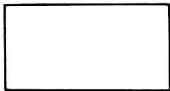
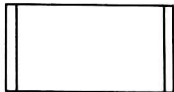
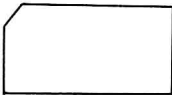
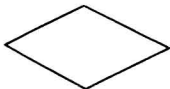
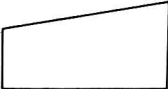
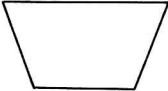
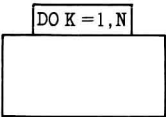
(記号)	(意味)	(記号)	(意味)
	処 理 一般的な処理機能		入 出 力 一般的な入出力処理機能.
	準 備 プログラム実行のための準備機能.		印 刷 印刷機能.
	定義済み処理 サブプログラムなどのほか処理機能の使用.		カード入力 カードの読取り機能.
	判 断 複数の経路の選択機能.		手操作入力 コンソールなどからのけん盤入力.
	端 子 開始, 終了, 中断などの端子.		手 作 業 機械的作業以外の手作業.
	結 合 子 フローチャートの結合.		くりかえし

図 0-10 フローチャート用の記号

チャートでは一般に□を入れる場所によって全体が複雑になったり、簡単になったりするのである。そこでフローチャートを書くには全体としていかに簡単になるかを考えながら□とか◇の置くべき場所を考えねばならない(しかしフローチャートが簡単になっても、それに対する計算機の計算時間が長くなる場合もあるので注意する必要がある)。

これまでに述べたフローチャートはすべて、「AからCへ行く」方法についてであったが、「CからA」と「BからD」などはどのようなフローチャートになるであろうか。実はこれは大きな問題ではない。始めに導入した記号は、それぞれの角に付けたものであり、Aはそこになければならないという理由は1つもない。したがって、今までBといていたところをAとし、CをB、DをC、AをDとすれば「BからDへ行く」方法は、すでに論じられたことになる。これも記号を導入し、抽象化したために得ることのできた長所である。

以上のようにある1つの仕事を分解してみると、簡単な行為からなっていることがわかる。しかし分解しただけではそれらの行為の前後関係がわからないので図にするわけである。

計算機で行なう仕事も複雑なものがあるが、それを計算機のできる小さな部分に分解し、夫々の部分の順序関係を線で結んだものをフローチャートという。

0-3 言語の型式

われわれが言語を用いる場合は、当然のことながらその言語の規則に従っている。この規則(文法)に従わないで書いたり話したりしても自分の意志を伝えることはできない。しかしながら言語にはいわゆる文法書に書いてある以前の規則ないしは習慣がある。たとえば

Im Anfang war das Wort, und das Wort
war bei Gott, und Gott war das Wort.

(はじめに言があった。言は神と共にあった。言は神であった。)

これは新訳聖書のヨハネ伝の一節であるが、ここで問題としようとしている