

1

アルゴリズム入門

1 アルゴリズムの定義

- 「明確に定義された有限個の規則の集まりであって、有限回適用することにより問題を解くもの」(JIS規格より)
- アルゴリズムを示すために、流れ図を用いる。

コンピュータで演算を行うには、その手順を教える必要があるんだ。それが「アルゴリズム」さ。

アルゴリズムによって、計算速度や精度が異なるから、気を付けないとね。



2 JIS流れ図の種類

①データ流れ図

- 問題解決におけるデータの経路を表し、かつ使用する各種のデータ媒体と共に、処理手順を定義する。
- 現在は、データフローダイアグラム（DFD）を用いることが多い（330ページ）。

②プログラム流れ図

- プログラム中における一連の演算を表す。
- プログラムの処理手順を示したもので、単に流れ図（フローチャート）という場合は、プログラム流れ図を指すことが多い。

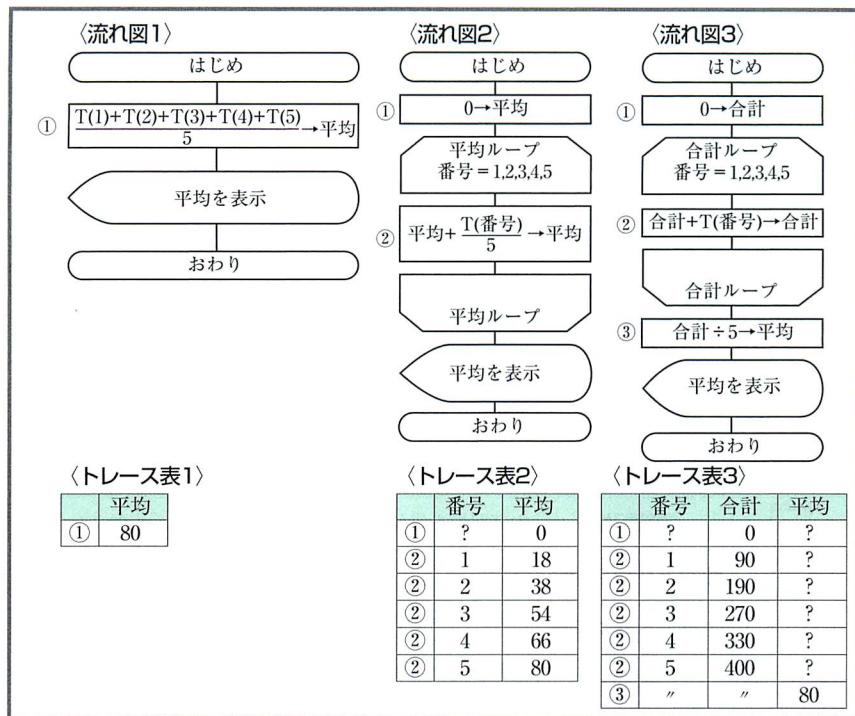
③システム流れ図

- システムの演算の制御やデータの流れを表す。
- データフローダイアグラムを用いることも多い。
- システムをプログラムに分割して、プログラム単位に処理の流れを示した図特にプロセスチャート（プロセス流れ図）と呼ぶ（354ページ）。

3 プログラム流れ図のトレース

- 5人分のテストの点数の平均点を求める。
- 配列Tに次のような得点が格納されているとする。

T (1)	T (2)	T (3)	T (4)	T (5)
配列T	90	100	80	60



1 アルゴリズムは、プログラム命令に対応する規則の集まりだ

アルゴリズムとは、あらかじめ定めた順序に従って実行すると、必ず終了する作業手順のことです。コンピュータでは、プログラムから手順の部分を抽出した規則（プログラムの命令文）の集まりといえます。

アルゴリズムを示すために、基本情報技術者試験では流れ図（フローチャート）を用います。一般の企業では、流れ図よりも、他の図式が用いられることが多くなっています。

2 同じ目的に複数のアルゴリズムがあり、計算時間や誤差が異なる

テストの平均点を求める手順にも、いくつかのアルゴリズムがあります。流れ図1は、一気に計算しています。データが5件程度なら最適なアルゴリズムですが、データが100件になると現実的ではありません。流れ図2は、得点を5で割ったものを足しています。このデータでは正確に求まりますが、データによっては正しい値が求まりません。たとえば、データ件数が3件で、得点が100, 100, 100のときは、平均点は100です。しかし、流れ図2で計算すると、100を3で割ることになるため、誤差が発生します。流れ図3がコンピュータでは一般的なアルゴリズムです。ループでまず合計を求め、データ件数で割ることで平均点を求めています。

第4章は、ご質問が多いので、ネットでダウンロードできる動画解説を用意しました。流れ図入門から解説しています。詳しくは、巻末を見てください。

2 流れ図記号

1 データ記号

① 基本データ記号

記号	名称	説明	デ	ブ	シ
	データ	媒体を指定しないデータを表す。	●	●	●
	記憶データ	処理に適した形で記憶されているデータを表す。媒体は指定しない。	●		●

② 個別データ記号

記号	名称	説明	デ	ブ	シ
	内部記憶	内部記憶を媒体とするデータを表す。	●		●
	順次アクセス記憶	順次アクセスだけ可能なデータを表す。 媒体は、磁気テープなど。	●		●
	直接アクセス記憶	直接アクセス可能なデータを表す。媒体は、磁気ディスクなど。	●		●
	書類	人間の読める媒体上のデータを表す。媒体は、プリンタで紙に印字されたものなど。	●	▲	●
	手操作入力	手で操作して情報を入力する、あらゆる種類の媒体上のデータを表す。キーボードなど。	●		●
	カード	カードを媒体とするデータを表す。せん孔カード、磁気カードなど。	●		●
	せん孔テープ	せん孔テープを媒体とするデータを表す。	●		●
	表示	情報を表示するあらゆる種類のデータを表す。ディスプレイなど。	●	▲	●

2 処理記号

① 基本処理記号

記号	名称	説明	デ	ブ	シ
	処理	任意の種類の処理機能を表す。	●	●	●

② 個別処理記号

記号	名称	説明	デ	ブ	シ
	定義済み処理	サブルーチンなど、別の場所で定義された処理を表す。	●		●
	手作業	人手による任意の種類の処理を表す。	●		●
	準備	その後の動作に影響を与えるための命令、または、命令群の修飾を表す。	●	●	●
	判断	1つの入口と択一的な出口を持ち、記号中に定義された条件の評価にしたがって、唯一の出口を選ぶ機能または、スイッチ形の機能を表す。		●	●

記号	名称	説明	デ	ブ	シ
=====	並列処理	2つ以上の平行した処理を同期させることを表す。		●	●
□□□	ループ端 ループ始端 ループ終端	始端と終端からなり、ループの始まりと終わりを表す。始端と終端は同じ名前を持つ。始端、または、終端の記号の中に、初期化、増分、終了条件を表記する。		●	●

3 線記号

① 基本線記号

記号	名称	説明	デ	ブ	シ
—	線	データ、または、制御の流れを表す。流れの向きを明示する必要があるときは、矢印をつけなければならない。	●	●	●

② 個別線記号

記号	名称	説明	デ	ブ	シ
△▽	制御移行	1つの処理から他の処理へ制御が即時に移行することを表し、場合によっては、起動された処理が終了した後に、起動させた処理に直接復帰することも表す。			
— —	通信	通信によって、データを転送することを表す。	●		●
-----	破線	2つ以上の記号の間の択一的な関係を表す。	●	●	●

4 特殊記号

記号	名称	説明	デ	ブ	シ
○	結合子	同じ流れ図中の他の部分への出口、または、他の部分からの入口を表したり、線を中断し他の場所に続けたりするのに用いる。対応する結合子は同一の一意な名前を含まなければならない。名前には、通常英字や数字を用いる。	●	●	●
□□□	端子	外部環境への出口、または、外部環境からの入口を表す。	●	●	●
-----[]	注釈	処理を明確にするために、説明または注を付加するのに用いる。	●	●	●
-----	省略	図の中で記号の種類も個数も示す必要がない場合に、記号または記号の集まりが省略されたことを示し、線記号に対してだけ用いる。	●	●	●

(注) デ: データ流れ図、ブ: プログラム流れ図、シ: システム流れ図で用いるものに●を付けた。JIS規格に厳密に従うと、「書類」「表示」記号をプログラム流れ図で用いることはできない。しかし、一般に「プリンタによる印刷」、「ディスプレイへの表示」の意味で用いることが多いため▲を付けた。「制御移行」は、この3つの図式では用いられない。

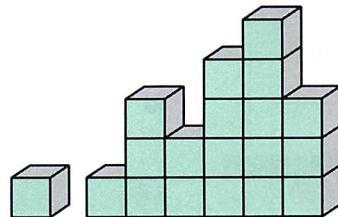
プログラムの処理手順を示すのは、プログラム流れ図です。プログラム流れ図で用いることができる記号をよく見ておきましょう。

3

構造化チャート

1 整構造プログラム

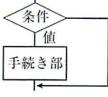
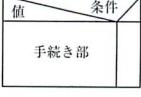
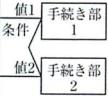
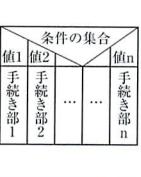
- 規定されたプログラム構成要素を適宜組み合わせて表現したプログラム。
- 一般には、**構造化プログラム**という。



2 プログラム構成要素と構造化チャート

- 整構造プログラムを図式化するために、流れ図以外のチャートがいくつか用いられている。これらを総称して**構造化チャート**と呼び、これらを使うと構造化（整構造）プログラムを表現しやすいという特徴がある。
- 手続部**…………1つ以上の実行される演算。一般には、**処理**と呼ぶ。
- 制御部**…………指示といくつかの条件からなり、手続部を実行する方法を指定する。

	プログラム構成要素	流れ図	N-Sチャート	PAD
基本	1つの手続部からなり、その手続部を1回だけ実行する。	<pre> +----+ 手續部 +----+ </pre>	<pre> +----+ 手續部 +----+ </pre>	<pre> +----+ 手續部 +----+ </pre>
順次	2つ以上の手続部からなり、その手続部を1回だけ実行する。	<pre> +----+ 手續部1 +----+ 手續部2 +----+ </pre>	<pre> +----+ 手續部1 +----+ 手續部2 +----+ </pre>	<pre> +----+ 手續部1 +----+ 手續部2 +----+ </pre>
繰返し	①前判定繰返し 1つの手続部と1つの条件を持つ制御部からなる。 条件を繰返しの前で判定し、その条件の値によって手続部を0回以上実行する。	<pre> +----+ 制御 +----+ 手續部 +----+ +----+ </pre>	<pre> +----+ 制御 +----+ 手續部 +----+ </pre>	<pre> +----+-----+ 制御 手續部 +----+-----+ </pre>
	②後判定繰返し 1つの手続部と1つの条件を持つ制御部からなる。 条件を繰返しの後で判定し、その条件の値によって手続部を1回以上実行する。	<pre> +----+ +----+ 手續部 +----+ 制御 +----+ </pre>	<pre> +----+ 手續部 +----+ +----+ </pre>	<pre> +----+-----+ 制御 手續部 +----+-----+ </pre>

プログラム構成要素	流れ図	N-Sチャート	PAD
①単岐選択 1つの手続部と1つの条件を持つ制御部からなる。 この条件は、手続部を実行するかどうかを指定する。			
②双岐選択 2つの手続部と1つの条件を持つ制御部からなる。 この条件は、2つの手続部のうち、どちらを実行するかを指定する。			
③多岐選択 多数の手続部と条件の集合を持つ制御部からなる。 これらの条件の値によって、実行すべき手続部が1つだけ決まる			

1 プログラムは、要素の組み合わせから構成される

わかりやすく、バグ（不具合）の少ないプログラムの開発手法として、GOTO文をなるべく使わず、順次、選択、繰返しの3つの基本構造で構成する構造化プログラミングが1980年代に注目され、現在では定着しています。これをJIS化したのが「プログラム構成要素及びその表記法」です。整構造プログラムとも呼ばれ、試験でもこの用語が用いられています。

2 流れ図は個人差が出やすい欠点がある

構造化プログラムで、プログラムがわかりやすくなりますが、流れ図を使う限り、GOTO文が必要になったり、同じプログラムでも書く人によって異なる流れ図になるという欠点が目立ってきました。そこで、流れ図に変わる図式として、N-Sチャート（Nassi-Shneiderman chart）やPAD（Problem Analysis Diagram）などの図式が開発されました。

余談ですが、著者が駆け出しのプログラマだった1980年代前半が、ちょうど流れ図からPADへの移行期でした。



所属する組織が採用している図式に精通すればよいわけで、すべて覚える必要はありません。試験では、N-Sチャートに関する問題が数回出題されています。