

第5章 プログラミングの基礎

5.1 アルゴリズム

ある問題を解決するための手段を**アルゴリズム**¹(algorithm) といいます。実は、すでに紹介した10進数を r 進数に変換した手順やシフト演算と足し算のみで掛け算や割り算を実行する手順もアルゴリズムです。以下の2つの問題のアルゴリズムを考えてみましょう。

最初の問題として、正の整数の平方根を求めるアルゴリズムを考えてみましょう。奇数の数列の和は

$$\sum_{i=1}^n (2i-1) = 1 + 3 + \cdots + (2n-1) = n^2$$

で与えられ、これを利用すると正の整数 $N (= n^2)$ の平方根 $\sqrt{N}$ の近似値を求めることができます。例として、5 ($= N$) の平方根 $\sqrt{5}$ の近似値を求めてみましょう。

$$5 = (1 + 3) + 1 = \{1 + (2 \cdot 2 - 1)\} + 1$$

ですから、5 の平方根の整数部分は2となります。小数点以下第1位まで求めるには、5 を100倍して ($\because 100N = n^2 \iff \sqrt{100N} = \sqrt{n^2} \iff 10\sqrt{N} = n \iff \sqrt{N} = n \div 10$)

$$\begin{aligned} 500 &= 20^2 + 100 = (1 + 3 + \cdots + 39) + 41 + 43 + 16 \\ &= \{(1 + 3 + \cdots + 39 + 41 + (2 \cdot 22 - 1))\} + 16 \end{aligned}$$

と変形することで、22 を10で割れば5の平方根の近似値2.2が得られます。

¹紀元9世紀、バグダッドのアル・フワーズリズミーという数学者が書き著した「キターブ・アルジャブル・ワ・ルムカーバラ (移項と消去の計算書)」という書名に由来して、algebra (代数学) という言葉が生まれました。また、この数学者の名前 アル・フワーズリズミーが変化してアルゴリズムという言葉になったといわれています。

更に、小数点以下第2位まで求めるには、5を10000倍して ($\because 10000N = n^2 \iff \sqrt{10000N} = \sqrt{n^2} \iff 100\sqrt{N} = n \iff \sqrt{N} = n \div 100$)

$$\begin{aligned} 50000 &= 220^2 + 1600 = (1 + 3 + \cdots + 439) + 441 + 443 + 445 + 271 \\ &= \{(1 + 3 + \cdots + 441 + 443 + (2 \cdot 223 - 1))\} + 271 \end{aligned}$$

と変形することで、223を100で割れば5の平方根の近似値2.23が得られます。同様に、この操作を繰り返せば5の平方根の近似値を必要な桁数まで求めることができます。

2つ目の問題として、正の整数 x, y の最大公約数 (greatest common divisor) を求めるアルゴリズムを考えてみましょう。ただし、 $x \geq y$ とします。初等的な代数を学習したみなさんであれば、**ユークリッドの互除法**という有名なアルゴリズムがあることをよく知っていて、

$$\gcd(x, y) = \begin{cases} x & , y = 0 \text{ のとき,} \\ \gcd(y, x \bmod y) & , y \neq 0 \text{ のとき} \end{cases}$$

で与えられたとき、 $y = 0$ になるまで繰り返し $\gcd$ を計算すれば最大公約数が求められます。ただし、 $x \bmod y$ は x を y で割った余りを表します。また、正の整数 x, y の最小公倍数 (least common multiple) は

$$\text{lcm}(x, y) = (x \times y) / \gcd(x, y)$$

を計算することで求められます。例として、1234と56の最大公約数と最小公倍数を求めてみましょう。上記の関係式を用いると、最大公約数は

$$\gcd(1234, 56) = \gcd(56, 2) = \gcd(2, 0) = 2$$

となり、最小公倍数は

$$\text{lcm}(1234, 56) = (1234 \times 56) / \gcd(1234, 56) = 69104 / 2 = 34552$$

となります。

問題 1 $\sum_{i=1}^n 2i - 1 = n^2$ を使って、7の平方根を小数点以下第3位まで求めなさい。

問題 2 ユークリッドの互除法を使って、3235と875の最大公約数と最小公倍数を求めなさい。

問題 3 正の整数 x, y, z ($x \geq y \geq z$) の最大公約数と最小公倍数はどのようにすれば求められるか考えなさい。

5.2 流れ図

流れ図(flow chart) はアルゴリズムを図式化したもので、コンピュータの処理手順となるデータの流れ・判定条件・実行の推移などについて流れ図記号²を用いて表現します。流れ図のようにアルゴリズムを図式化することは、問題の定義や分析または解法をより明確にするため、プログラム作成の大きな手助けとなります。また、第三者にも正確に伝えることができます。

流れ図について詳しく説明する前に、流れ図やプログラムで用いられる変数と比較の記述について注意しなければならないことがあります。コンピュータは、変数と呼ばれる数値を格納する箱の値の変化と変数の比較によってプログラムを実行していきます。流れ図の場合には、変数 x に 1 を代入するには「 $x \leftarrow 1$ 」と記述し、 x と 1 を比較するには「 $x : 1$ 」に対して条件「 $> (<, =, \leq, \geq)$ 」を記述します。また、プログラミング言語の多くは、代入を「 $x = 1$ 」のように記述するため、数学で比較を表す「 $x = 1$ (x と 1 が等しい)」と勘違いしないようにしてください。C 言語では「 $==$ (右辺と左辺が等しいならば)」が用いられます。なお、流れ図でもプログラミング言語の書き方で記述する場合があるので適切に読み替えてください。

変数について、もう一つの注意点は $x \leftarrow x + 1$ や $x = x + 1$ と記述されている場合です。これは、変数 x の値に 1 を足して x に代入するという意味です。例えば、変数 x の値が 2 であったとすると、 $x \leftarrow x + 1$ や $x = x + 1$ を実行すると x の値は 3 ($= 2 + 1$) になります。 $x = x + 1$ を数学の等式と勘違いすると $0 = 1$ といった奇妙な状態になるので間違えないようにしてください。

それでは、流れ図について詳しく見ていきましょう。流れ図に用いられる主な記号には図 5.1 の左上から下に順に、端子・接合子・入力・準備・処理・定義済み処理・ループ端上・ループ端下・判断を表します。流れ図はこれらの記号を線で繋ぐことで作成されます。

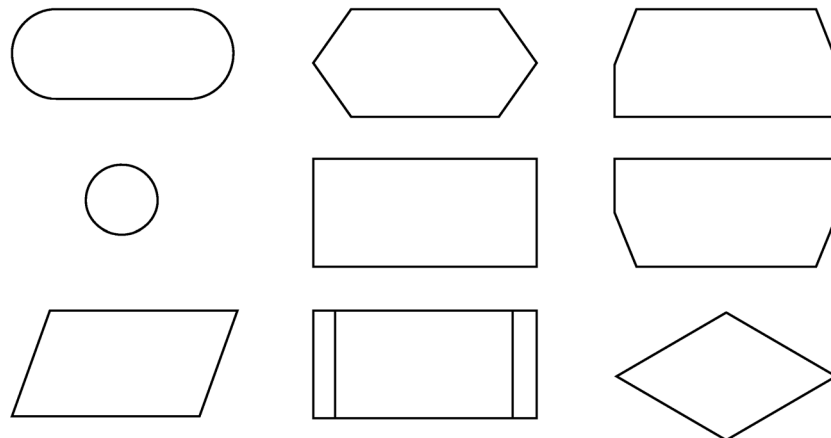


図 5.1: 流れ図記号

²このテキストでは JIS X 0121(1986) 情報処理用流れ図記号を用います。

前節のアルゴリズムを流れ図にすると次のようになります。

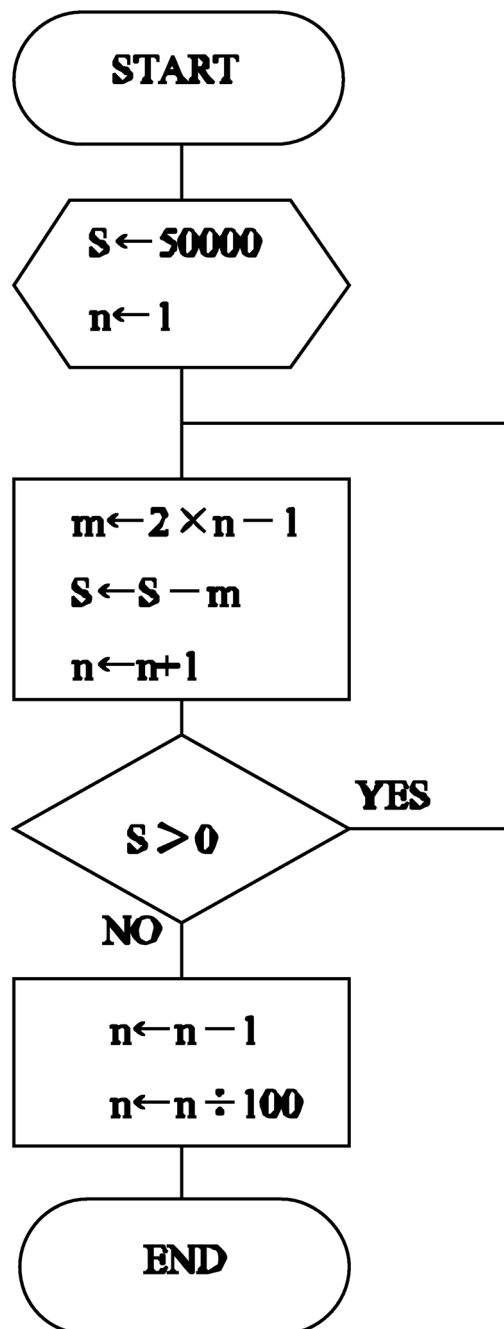
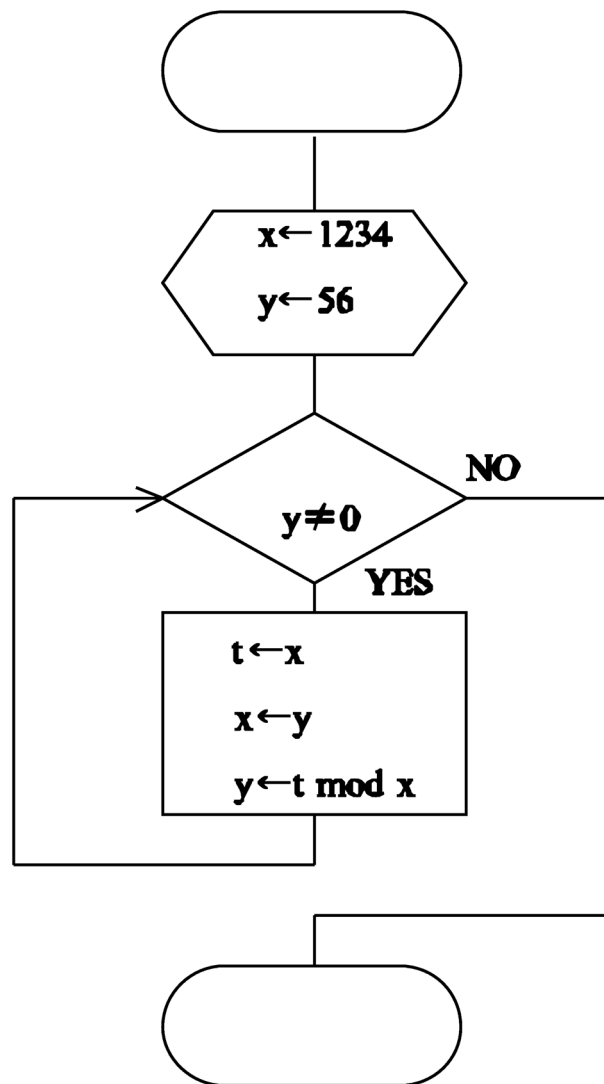


図 5.2: 5 の平方根 2.23 を求める流れ図

図 5.3: 正の整数 x, y の最大公約数を求める流れ図

問題 1 1 から 50 までの和を求める流れ図を作成しなさい。

問題 2 最小公倍数を求める流れ図を作成しなさい。