

# 関数の情報受け渡し… 「引数」と「戻り値」

鹿取 祐二

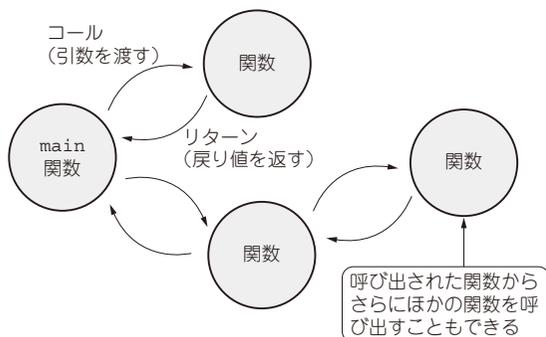


図1 関数が複数あるプログラムの構造

main関数から機能単位に分割された関数を呼び出している様子。機能を細分化すると構造が分かりやすくなる

特設 第1章でC言語は、

- 関数がプログラムの最小構成単位であること
- 最初に実行されるのはmain関数であること
- main関数はvoid main(void)の形式で記述すること

などを紹介しました。

本章では、複数の関数で1つのシステムを作るときに必要な関数の引数と戻り値について解説します。

## ● 機能が細分化されていると構造が分かりやすい

複数の関数で1つのシステムを構築する目的は、システムが持つ機能を細分化して、分かりやすい構造にすることです。

図1に示すように、main関数から機能単位に分割された関数を呼び出します。呼び出された関数も、必要に応じてほかの関数を呼び出すことで、システムの処理を分かりやすく記述できます。細分化された関数は、他のシステムで再利用できるので、コーディング時間の短縮につながる場合もあります。

## ● 「引数」と「戻り値」で情報を受け渡す

C言語では、関数を呼び出すときに目的の関数に渡す情報(処理を依頼するときのパラメータ)のことを

リスト1 2つの整数値の最大公約数を求めるプログラム(第4部第2章のリスト6の再掲)

本章ではこのコードをベースに最大公約数を求める処理を関数として記述してみる

```

int r;
while( b != 0 ) {
    r = a % b;
    a = b;
    b = r;
}

```

引数と呼びます。逆に、呼び出し元の関数に返す情報(処理結果のパラメータ)のことを戻り値とか返却値と呼びます。ここでは、引数や戻り値を受け渡す方法を紹介します。

リスト1に示す、while文を使って2つの整数値の最大公約数を求めるプログラムを例に、本章では最大公約数を求める処理を関数として記述してみます。

### 1 関数の呼び出し方と引数の渡し方 (呼び出し側)

細分化された関数の呼び出しは、次のように記述します。

関数名(引数の列);

呼び出す関数の名前である関数名を記述し、その後の()内に目的の関数を渡す引数の列を記述したら、最後に;を付けます。引数の列には、式の記述が可能で、引数が複数ある場合はカンマ(,)で区切って記述します。

例えば、最大公約数を求める関数を関数名gcmとした場合、最大公約数を計算する元の情報としては2つの整数値が必要なので、直接定数値で記述するなら次のようになります。

```

gcm( 24, 18 );
// 定数値を引数とした関数呼び出し

```

この呼び出し元に記述された引数は、実引数とも呼ばれます。実引数には式を記述できます。式なので、変数を使えます。また、変数や値を演算子で結合した記述を使っても構いません。