

入力処理, 条件判断処理

前回は printf による標準出力と変数の宣言と利用方法に触れた。詳しくはもう少し後で習う。今回も C 言語のプログラミングがどういったものかを知るため、入力処理と条件判断処理を習っていく。

入力処理

C 言語の標準入力にはキーボードが割り当てられている。ここではキーボードから整数の値を入力する方法を示す。キーボードから値を入力するには scanf 文を用いる。まずはプログラム例を示していく。

| プログラム例 1 |
|--|
| <pre>#include <stdio.h> int main(void) { int x, y; printf("x と y で演算を行います\n"); printf("x の値:"); scanf("%d", &x); printf("y の値:"); scanf("%d", &y); printf("x + y = %d\n", x + y); printf("x - y = %d\n", x - y); printf("x * y = %d\n", x * y); printf("x / y = %d\n", x / y); printf("x %% y = %d\n", x % y); return 0; }</pre> |
| 出力例(斜体はキーボードから入力した値) |
| <pre>x と y で演算を行います x の値: <i>9</i> y の値: <i>4</i> x + y = 13 x - y = 5 x * y = 36 x / y = 2 x % y = 1</pre> |

値を入力するには、変数を使う必要がある。変数を宣言しておき、以下の書式で scanf 文を利用する。これは printf 文で整数を出力する場合と似ている。大事な点は変数名の前に「&」を書くことである。

| 書式 |
|--|
| <pre>int 変数名; scanf("%d", &変数名);</pre> |

条件判断処理

問題を解決するための処理手順をアルゴリズムという。コンピュータに仕事をさせるためにはまず、アルゴリズムを考え、プログラミングを行い、コンパイラを用いて機械語に翻訳してコンピュータに実行させる必要がある。アルゴリズムには大きく分けて3つの処理の流れがある。それらは「順次処理」、「条件判断処理」、「ループ処理」という。

ここまでの処理は上から順に実行される「順次処理」であった。今回は「条件判断処理」のプログラムを習う。プログラム例1では変数を用いて割り算を行っている。数学の割り算では分母が0の場合、結果は無限大になる。コンピュータではある程度大きな数を利用することができるが、限界があり、無限大を表すことはできない。このプログラムの場合、yの値に0を入力してしまうと正しく動作しなくなる。これは剰余を求める時も同様である。

この問題を回避するには割り算と剰余の計算を行う際にyの値をチェックし、0だった場合には「計算できません」と表示するように変更する。

条件判断処理ではif-else文を用いる。プログラムは図1のような形になる。if文の条件式と呼ばれる部分で条件の判定を行い、正しければ処理Aを実行し、正しくなければ、処理Bを実行する。

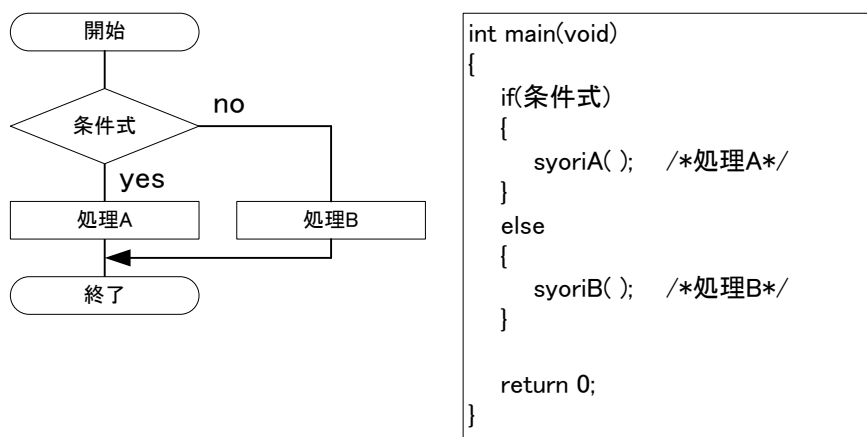


図1 条件判断処理

条件式では、= (等しい)、≠ (等しくない)、≥ (以上)、≤ (以下)、> (大なり)、< (小なり)などの関係を比較することができる。C言語ではそれぞれ「==」「!=」「>=」「<=」「>」「<」と書く。これらを総称して「関係演算子」という。関係演算子の一覧を表1に示す。これに対して計算に用いた「+」「-」「*」「/」「%」は「算術演算子」と呼ばれている。

表1 関係演算子一覧表

| 関係演算子 | 数学的表現 | 意味 | プログラムでの表現 | その意味 |
|-------|-------|-----|-----------|-----------|
| == | = | 等号 | a == b | aとbは等しい |
| != | ≠ | 不等号 | a != b | aとbは等しくない |
| >= | ≥ | 以上 | a >= b | aはb以上である |
| <= | ≤ | 以下 | a <= b | aはb以下である |
| > | > | 大なり | a > b | aはbよりも大きい |
| < | < | 小なり | a < b | aはbよりも小さい |

それでは条件判断処理を用いて、プログラム例1を書き直したものを以下に示す。

プログラム例2

```
#include <stdio.h>

int main(void)
{
    int x, y;

    printf("x と y で演算を行います\n");

    printf("x の値:");
    scanf("%d", &x);

    printf("y の値:");
    scanf("%d", &y);

    printf("x + y = %d\n", x + y);
    printf("x - y = %d\n", x - y);
    printf("x * y = %d\n", x * y);

    if(y == 0)
    {
        printf("割り算と剰余は計算できません\n");
    }
    else
    {
        printf("x / y = %d\n", x / y);
        printf("x %% y = %d\n", x % y);
    }

    return 0;
}
```

出力例(斜体はキーボードから入力した値)

```
x と y で演算を行います
x の値: 9
y の値: 0
x + y = 9
x - y = 9
x * y = 0
割り算と剰余は計算できません
```

if 文の分岐の判定は「y == 0」で行っており、変数 y の値が 0 と一致するかどうかを判定している。注意点は「=」を二つ連続して書くことである。「=」が一つの場合には代入が行われてしまい、行いたい処理が行われぬ。

練習として、もう一つの例を示す。以下のプログラム例は入力された値の絶対値を表示するプログラムである。

プログラム例3

```
#include <stdio.h>

int main(void)
{
```

```

int x;

printf("整数を入力してください:");
scanf("%d", &x);

if(x < 0)
{
    printf("絶対値は%d です\n", -x);
}
else
{
    printf("絶対値は%d です\n", x);
}

return 0;
}

```

出力例(斜体はキーボードから入力した値)

```

整数を入力してください:-9
絶対値は9です

```

演習

- ・テキストで示したプログラム例1～3を作成すること。
- ・次のプログラムに取り組むこと。
 - (1) 以下のような出力を行うプログラムを作成しなさい。

(注意) 計算結果を出力させる部分については変数を用いて計算し、出力させること。
□部分はスペースである。

| C | 言 | 語 | □ | 練 | 習 |
|---|---|---|---|---|---|
| 3 | + | 2 | = | 5 | |
| 9 | - | 7 | = | 2 | |

※前回のテキストで示した演習です。作成済みの人は(2)の演習に進みます。

- (2) プログラム例1を参考に電流と抵抗の値を入力し、オームの法則から電圧を求めて表示するプログラムを作成しなさい。入力する値、出力する値は整数値とする。プログラムを利用する人が迷わないように、こういった計算を行うのか、こういった値を入力するのか、処理の結果がどうなったかを出力すること。
- (3) プログラム例2の条件式を「 $y \neq 0$ 」に変更する。同様の動作を行う用に他の部分を修正しなさい。
- (4) プログラム例3の条件式を「 $x \geq 0$ 」に変更する。同様の動作を行う用に他の部分を修正しなさい。
- (5) 余裕がある人はここまで習ったことを使ってオリジナルのプログラムを作成すること。