

第3回 *if*文・*switch*文

プログラムを書く上では、「制御文」が非常に重要になります。制御文とは、C言語で決められている、プログラムの流れを制御するものです。これにより”ある条件”で違う流れのプログラムに切り替えたり、同じ処理をある回数だけ実行したり、ということを示すものです。今回は、条件によってプログラムの流れを変える方法を解説します。

if 文

まず、プログラムを分岐させる最も基本的な制御文が if 文です。if は、英語と同じで「もしも～なら」ということを意味します。

```
#include<stdio.h>

int main(void)
{
    int a;

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

    if(a>10){
        printf("その整数は 10 より大きい数です。");
    }
    return 0;
}
```

太字部分が if 文です。この部分は、

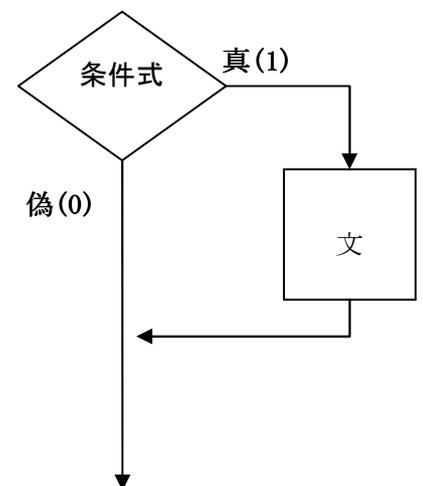
```
if (条件式){
    文;
}
```

という形になります。

条件式は、内容が指定された条件に合わなければ 0 を、合えば 1 を返します。この 0 のことを偽、1 のことを真といいます。

if 文では条件式が真か偽かを判断し、真ならば指定された文を実行させます。

上の文では、入力した値が 10 以上ならばその後の文を表示させます。

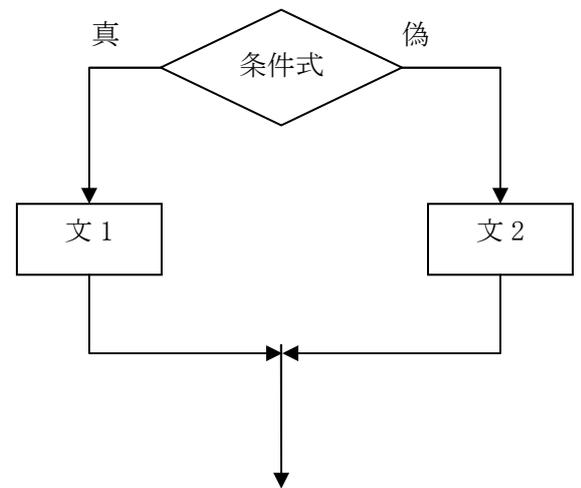


else

ところで、上のプログラムでは 10 より大きい値を入力した場合は文が表示されますが、10 以下の場合は何も表示されません。プログラムを使っている人が不安にならないために、それ以外の場合についても文を表示させる必要があります。そのために、上のプログラムの if 文の部分に **else** を用いて次のように書き換えます。

```
if(a>10){
    printf("その整数は 10 より大きい数です。");
}
else{
    printf("その整数は 10 以下です。");
}
```

else も英語と同じで「～でなければ」ということを意味します。こうすることにより、どちらの場合でも文を表示させることができます。



演算子

ところで、条件式では次のような「条件演算子」が使われます。

a<b	a は b より小さい
a<=b	a は b と同じか小さい
a>b	a は b より大きい
a>=b	a は b と同じか大きい
a==b	a は b と等しい
a!=b	a は b と等しくない
a&& b	a かつ b
a b	a または b

これらを使えば、様々な条件をつけることができます。

```
#include<stdio.h>

int main(void)
{
    int a;

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

    if(a>0){
        printf("その数は正です。");
    }
    else if(a<0){
        printf("その数は負です。");
    }
    else if(a==0){
        printf("その数は0です。");
    }

    return 0;
}
```

else if

上のプログラムは else if が使われています。プログラムを様々な方向に分岐させたいときは、このようにすると良いです。また、以下のように書き換えることもできます。

```
if(a>0){
    printf("その数は正です。¥n");
}
else if(a<0){
    printf("その数は負です。¥n");
}
else{
    printf("その数は0です。¥n");
}
```

switch 文

何方向にもプログラムを分岐させる場合には、その度に if 文を書き連ねる必要があります。しかし、これを一度に分岐させる方法があります。それが switch 文です。

```
#include <stdio.h>

int main(void)
{
    int a;

    printf( "整数を入力して下さい" );
    scanf( "%d", &a );

    if(a==0){
        printf("0 が入力されました\n");
    }
    else if(a==1){
        printf("1 が入力されました\n");
    }
    else if(a==2){
        printf("2 が入力されました\n");
    }
    else if(a==3){
        printf("3 が入力されました\n");
    }
    else if(a==4){
        printf("4 が入力されました\n");
    }
    else{
        printf("0~4 以外が入力されました\n");
    }
    return 0;
}
```

上の if 文の部分を switch 文にすると、以下のようになります。

```
switch( a )
{
case 0:
    printf( "0 が入力されました\n" );
    break;
case 1:
    printf( "1 が入力されました\n" );
    break;
case 2:
    printf( "2 が入力されました\n" );
    break;
case 3:
    printf( "3 が入力されました\n" );
    break;
case 4:
    printf( "4 が入力されました\n" );
    break;
default:
    printf( "0~4 以外が入力されました\n" );
    break;
}
```

このようにすれば、条件式をいちいち書く必要はなくなります。後から見直すときも少しだけ見やすくなります。

※ default は、どの case にも当てはまらない場合に向かう目印です。

※ break には、switch 文を終了させるはたらきがあります。これを書かないと、他の case も順に実行してしまいます。

```
switch( a )
{
case 0:
    printf( "0 が入力されました\n" );
    break;
case 1:
    printf( "1 が入力されました\n" );
    break;
case 2:
    printf( "2 が入力されました\n" );
case 3:
    printf( "3 が入力されました\n" );
    break;
case 4:
    printf( "4 が入力されました\n" );
    break;
default:
    printf( "0~4 以外が入力されました\n" );
    break;
}
```

上の例では、case 2 の部分だけ break がありません。このプログラムで 2 を入力すると、case 2 の後の case 3 も実行してしまい、

2 が入力されました

3 が入力されました

と表示されてしまいます。このようなことを防ぐためにも、break は必ずつけるようにしましょう。

問題

整数を 3 つ入力させて、どの数が一番大きいかを表示するプログラムに if 文を用いて作れ。