

配列

今までは変数を宣言するときに一つ一つの変数名を宣言してきました。この方法では大量の変数を宣言するときに、かなり大変ですよ。そんなときに使用するのが今回習う「配列」です。

♣配列とは？

配列は複数の変数を一気に管理したいときに利用します。配列の宣言の形を見て説明します。配列の宣言の仕方は、

```
データの型 配列名[配列の大きさ];
```

(↑intとか ↑変数名 ↑欲しい変数の個数)

という形になります。どのように使用するかというと、

```
int a[3];          // == int a[0],a[1],a[2];
```

と宣言したとすると、`a[0],a[1],a[2]`という3つの変数が宣言されたことになります。[]内の数字を添え字と呼び、欲しい変数の数を代入します。添え字は0から始まりますのでこの場合 `a[3]`という変数は宣言されたことにはなりません。次のソースコードを見てみましょう。

ソースコードの長さの都合上の余白
ソースコードは次のページです。

```

#include<stdio.h>
int main(void){
    int a[3],i;
    a[0]=2;
    a[1]=3;
    a[2]=5;
    printf("a[0] = %d\n",a[0]);
    printf("a[1] = %d\n",a[1]);
    printf("a[2] = %d\n",a[2]);
    printf("ここから for 文\n");
    for(i=0; i<3; i++){
        printf("a[i] = %d\n",a[i]);
    }
    return 0;
}

```

配列の添え字に変数を使用することで for 文で楽に入出力できるようになります。for 文で動く変数 i が 0 から始まるのは添え字が 0 から始まるからだったのです。また、配列の初期化は

```
int a[3]={2,3,5};
```

```
int a[3]={0}; //int a[3]={0,0,0};
```

とやります。

❖ 多次元配列

2次元配列は配列の配列です。3次元配列は配列の配列の配列です。

```
int a[3][3]; // 2次元配列 3*3の配列
```

```
int b[3][3][3]; // 3次元配列 3*3*3の配列
```

このように配列に[添え字]を足すことで多次元配列を作ることができます。初期化するには、

```
int a[3][3]={{1,2,3},{4,5,6},{7,8,9}};
```

```
int b[3][3][3]={{0}};
```

といった様を書くことで行えます。

♣switch 文

Nintendo Switch が発売されましたね。残念ながらこの switch 文とは関係ありません。switch 文とは条件分岐の方法として用いられるものの一種です。指定した変数の値を見て条件分岐をします。使い方は、

```
switch(変数){
    case 値1 :
        文1;
        break;
    case 値2 :
        文2;
        文3;
        break;
    default :
        文4;
        break;
}
```

というふうに使います。変数が各値と等しいときにそこにある文を実行します。指定した値以外のときにも文を実行したい場合は default を使います。break;は switch 文を抜け出すために必要で、これがないとそのまま下の case も実行してしまいます。

・サンプルプログラム

```
#include<stdio.h>
int main(void){
    int a;
    printf("入力 :");
    scanf("%d",&a);
    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;
        default : printf("入力した値は 0 と 1 ではありません。¥n");    break;
    }
    return 0;
}
```

❖ 練習問題

1. 4人の数学、英語、物理の成績を入力して各科目の平均点を出力させてください。
2. サイコロの出目を入力するとその目の反対側の数字が出力されるプログラムを作りなさい。