C言語講座
第4回

関数－FUNCTION－

関数(Function)

みんなが「関数」と聞いて思い浮かべるのは

$$y = f(x)$$

かな？

数学では“値を求める”為にしか使わなかったが、プログラミングでは“引数[()の中身]に応じて仕事をしてもらうもの”だ。

今までのサンプルに出てきた、

```
void main(void) { (中略;) }
```

も関数の一つ。

というか、C言語のプログラムは基本的にこの関数の本体部が実行されるので、

たいていは、これが無いとお話にならない。

このようなものをエントリーポイントと呼びます。

mainは特殊な関数な訳ですが。

C言語が提供するライブラリ関数と呼ばれるものがあります。

printf, scanf

とか。これ以外にも数え切れない？くらいあります。

関数定義と呼び出し

と言っても、C言語で定義された関数だけ使うわけにもいかず、しっかりしたプログラムを組みたいなら自分の用途にあった関数を作る必要がある。

○構文

```
返却値型  関数名(引数宣言)
{
    関数本体
}
```

○プログラミング実践

二つの整数の内、大きい方を返す

```
#include <stdio.h>
int maxInt(int a, int b) {
   if  (a  >=  b)  {  return   a;  }
   return  b;
}
void main(void)
{
   int max = maxInt(10,30);
   printf("大きい方の値:%d¥n", max);
}
```

〇解説

今回重要なのは”return”

これは、返却型に適した(↓ならint型)

値を返してくれるキーワード。

基本的には書かれてないといけない。

```
int maxInt(int a, int b) {
   if  (a  >=  b)  {  return   a;  }
   return  b;
}
```

○解説_続き

そして、その返ってきた値(↓ならint型)は、変数のように代入にも使える。

```
void main(void)
{
    int max = maxInt(10,30);
    printf("大きい方の値:%d¥n", max);
}
```

変数maxにmaxInt関数から最大値を代入し、printf関数で表示している。

関数原型(プロトタイプ)宣言

Ｃ言語の仕様として、“後方参照が出来ない”というものがある。

簡単に言うと、宣言してからじゃないと使えないよってこと。

試しに、前サンプルの関数２つの順番を入れ替えてみるとエラーが起きるはずだ。

だけど、関数の順番をいちいち気にしているのも面倒な話。

そんな時に使うのがプロトタイプ宣言。

プロトタイプ宣言

・・・関数の返却値型や、仮引数の型や個数を明示した宣言のこと。

○プログラミング実践

二つの整数の内、大きい方を返す

```
#include <stdio.h>
int maxInt(int a, int b);
void main(void)
{
   int max = maxInt(10,30);
   printf("大きい方の値:%d¥n", max);
}
int maxInt(int a, int b) {
   if  (a  >=  b)  {  return   a;  }
   return  b;
}
```

void

返却値型、っていうわりにmainの中に“return”が無いのはなぜだろうか。

これは“void”がキーワード。

voidは返すべき値を持たないときに用いられる特殊なワードで、英語では「空」「無」意味する…らしい。

○プログラミング実践

二つの整数の和を表示する

```
#include <stdio.h>
void showAdd(int a, int b) {
    printf("%d + %d = %d¥n", a, b, a+b);
}
void main(void)
{
    showAdd(5, 7);
}
```

当然、値を返さないので代入は出来ない

○演習

整数を入力、それに合わせて、下図のように表示するプログラムを作成してください。

ただし、main関数の中で出力関数は使わないこと。

例：5を入力したとき

```
*
**
***
****
*****
```

○解答例(次ページへ続く)

```
#include <stdio.h>
int userIn(void)  {
    int num;
    printf("整数を入力してください:");
    scanf("%d", &num);
    return num;
}
void putStars(int n) {
    int i;
    for (i = 0; i < n; i++) {
        printf("*");
    }
    printf("¥n");
}
```

○解答例_続き

```
void main(void)
{
    int i;
    int rank = userIn();
    for (i = 1; i <= rank; i++) {
        putStars(i);
    }
}
```

関数宣言ではないのでvoid不要