

# C言語講座

## 第一回

出入力(*printf*, *scanf*)

*int char*型 四則演算



# 新しいプロジェクトの作り方

- **1. Microsoft Visual studio 2010 C++を起動します。**
- **2. 「ツール」⇒「設定」⇒「上級者用の設定」を選択。**
- **3. 「新しいプロジェクト」⇒「win32コンソールアプリケーション」を選択し、名前をここでは「Lesson1」とします。**
- **4. アプリケーションウィザードがあるので、「次へ」⇒「空のアプリケーション」を選択し、「完了」します。**



# 新規ソースファイルの作成

- 1. 「ソースファイル」フォルダを右クリック。
- 2. 「追加」⇒「新しい項目」を選択。
- 3. 「C++ファイル」を選択し、名前を管理しやすい適当な名前にして(日本語はNG)、「追加」を選択。



# 文字を出力してみよう

```
#include <stdio.h>

void main(void){
    printf("Hello World¥n");
}
```

「デバック」⇒「デバックなしで開始」で実行(ctrl+F5)



# 表示結果

```
Hello World  
続行するには何かキーを押してください...
```

**と表示されるはずですが。**

**表示されなかった場合はプログラムを見直してください。**

**「例)「:」が抜けていないか、includeの前に#をつけ忘れていないかなど...**

**見直してもわからなかったら先輩に聞いてください。**



# ソースの解説

- `#include<stdio.h>`や`void main()`の説明は少し難しいので、今は呪文のように覚えてください。
- 詳しくは今後の講座でほかの先輩が説明します。
- `printf`とは” ”で囲まれた文字、もしくは文字列を出力するものです。

```
printf(“この文字（文字列）を出力”);
```

# 拡張表記 (エスケープシーケンス)

- “¥n” のことを拡張表記またはエスケープシーケンスといいます。
- 拡張表記とはプログラミング言語で文字としてではなく、意味のある記号として認識されるものです。
- 拡張表記は“¥n”のほかにもいろいろあります。

¥n	改行
¥t	タブ
¥¥	¥を文字として使いたいとき

などなど...

# 変数と変換指定子

```
#include <stdio.h>

void main(void){
    int a;
    a = 10;
    printf("a= %d", a);
}
```



# 表示結果

a=10

続行するには何かキーを押してください...

**と表示されるはずですが。  
表示されなかった場合は、さっきのようにプログラム  
を見直してください。**



# 変数

- 変数とは、数字などを入れておくための箱のようなものだと思ってください。
- 注)変数を使うためには事前に宣言しないとイケません。
- さっきのプログラミングではaを変数として使いたかったので、`int a;`と事前に宣言しています。
- 変数には以下のようなものがあります。

# 変数の基本型

基本形	型	範囲
int	整数型	-2,147,483,648～2,147,483,647
short	整数型	-32,768～ 32,767
long	整数型	-9,223,372,036,854,775,808 ～ 9,223,372,036,854,775,807
char	文字型	1文字
float	浮動小数点型	$-1.79769313486231570 \times 10^{308}$ ～ $1.79769313486231570 \times 10^{308}$
double	浮動小数点型	$-1.175494 \times 10^{38}$ ～ $1.175494 \times 10^{38}$

# 変数の基本型

基本形	型	範囲
int	整数型	-2,147,483,648 ~ 2,147,483,647
short	整数型	-32,768 ~ 32,767
long	整数型	-9,223,372,036,854,775,808 ~ 9,223,372,036,854,775,807
char	文字型	1文字
float	浮動小数点型	$-1.79769313486231570 \times 10^{308}$ ~ $1.79769313486231570 \times 10^{308}$
double	浮動小数点型	$-1.175494 \times 10^{38}$ ~ $1.175494 \times 10^{38}$

# 変換指定子

- “%d” のような “d” のことを変換指定子と呼びます。
- 文字以外のものを文字に変換する機能を持っていて、その機能は “%” に続く文字によって異なります。

%d	整数を10進数で表示
%o	整数を8進数で表示
%x	整数を16進数で表示
%lf	実数を小数点つき10進数で表示
%c	引数に対応した1文字を表示
%5d	整数を10進数で表示かつ5桁で表示。数字はいくつでもよい。（空いているところは空白になる）
%05d	整数を10進数で表示かつ5桁で表示。数字はいくつでもよい。（空いているところは0で表示）

# 演算子

- 加減演算子

$a+b$	aとbの和
$a-b$	aとbの差

- 乗除演算子

$a*b$	aとbの積
$a/b$	aをbで割った商（整数どうしの場合小数点以下は切り捨て）
$a\%b$	aをbで割った余り（aとbは整数でないといけない）

# 変数と変換指定子②

```
#include<stdio.h>

void main(void){
    int a,b,c;
    a = 10;
    b = 20;
    c = a+b;
    printf(“%d+%d=%dです。 ¥n”,a,b,c);
}
```

# 表示結果

10+20=30です。

続行するには何かキーを押してください...



# 入力

```
#include <stdio.h>

void main(void){
    int a;
    printf(“整数を入力してください⇒” );
    scanf(“%d”,&a);
    printf(“入力した数字は%dです。¥n”,a);
}
```

**コンソール画面に数字を入力する画面が出るので、自分で入力してEnterを押してください。**

# 表示結果

整数を入力してください⇒10

入力した数字は10です。

続行するには何かキーを押してください. . . .

**これは10を入力した例です。**

**他の数字も自分でいろいろ試してみてください。**

## ソースの解説②

- **scanfとはキーボードから数値などを読み込むために用いる関数のことです。**
- **気を付ける点は、”&”を付けること。なぜ”&”を付けるのかは第5回のポインタの講義で説明するので今はおまじない程度に…**



# Character型

```
#include<stdio.h>

void main(void){
    char a;
    printf("一文字入力してください⇒");
    scanf("%c",&a);
    printf("入力された文字は「%c」です。¥n",a);
}
```

# 表示結果

一文字入力してください⇒a

入力した文字は「a」です。

続行するには何かキーを押してください. . .

**これはaを入力した例です。**

**他の文字も自分でいろいろ試してみてください。**

# Character型

- **char型は文字を記憶するための変数です。**

```
char a = 'a';
```

- **注)文字を表すためには「 ' 」を使います。また、char型は**1文字**しか表せません。**



# 演習①

もしもし。  
こんにちは。

それでは。

と表示されるプログラムを作ってください。  
ただし、**printf**を使うのは**1回のみ**！

## 演習②

二つの変数を用意して、それらにscanfで10進数を入力してから、

$a+b$   $a-b$   $a*b$   $a/b$   $a\%b$

を画面に表示させてください。

(ただし、 $b$ が0であることを考慮しなくてよい。

また、 $\%$ を文字で出力する場合は $\%\%$ と書く。)



**次のページに  
答えがあります**



# 解答例①

```
#include <stdio.h>
```

```
void main(void){
```

```
    printf("もしもし。¥nこんにちは。¥n¥nそれ  
では。¥n");
```

```
}
```

# 解答例②

```
#include <stdio.h>

void main(void){
    int a,b;
    printf("a=" ); scanf("%d", &a);
    printf("b=" ); scanf("%d," &b);
    printf("a+b=%d¥n", a+b);
    printf("a-b=%d¥n", a-b);
    printf("a*b=%d¥n", a*b);
    printf("a/b=%d¥n", a/b);
    printf("a%%b=%d¥n", a%b);
}
```