C言語講座第2回

作成:ハルト

○ 前回の復習

基本的に main() { } の中カッコの中にプログラムを書く。 また、変数(int, float)は C 言語では main() { } の中カッコの先頭で宣言する。

```
→printf()
② キーボードから入力
→scanf()
※ printf / scanf で
整数を表示/入力
→%d
小数を表示/入力
→%f
```

① 画面へ出力

- ③ 整数を扱う→int 型を使う④ 小数を扱う
- ④ 小数を扱う→float 型を使う

○ 条件分岐(if 文、switch 文)

```
if 文書き方:
if(条件式){
//処理1}
else
//処理2
動作:
条件式の内容が正しかったらif(){} の中の処理1を行う。そうでなかったらelse{} の中の処理2を行う。else は省略可能
```

```
· switch 文
書き方:
switch (式)
   case 0:
      //処理 1
   case 1:
      //処理 2
   default:
   //処理 n
}
動作:
switch(式)の式の値を調べる。
その値が case のどれかと等し
い場合その case に書かれてい
る処理を行う。
どれとも一致しない場合
default に書かれている処理を
行う。
なお、default は省略可能。

    break

switch 文から抜ける。
```

これを書かないと次のcaseの処

理まで実行されてしまう、

```
#include <stdio.h>
int main(void)
       int num_a , num_b;
                            //計算元の数値
                            //行う計算
       int sentaku;
                            //計算結果
       int result = 0;
       printf("行う処理は?\n");
       printf("1. 加算 2. 減算 3. 乗算 4. 除算\n->");
       scanf("%d", &sentaku );
       printf("整数値を2つ入力してください\n");
       scanf ("%d %d", &num a , &num b ); //2つ値を入力してもらう
       switch( sentaku )
       case 1:
                                   //加算
              result = num_a + num_b;
              break;
       case 2:
                                   //減算
              result = num_a - num_b;
              break:
       case 3:
                                   //乗算
              result = num_a * num_b;
              break;
       case 4:
                                   //除算
              result = num_a / num_b;
              break:
                                   //1~4以外の値が入力された
       default:
              printf("不正な選択肢です\n");
              break;
       printf("計算結果は%dです\n", result);
       return 0;
```

○ループ文(for , while)

```
特定の回数だけループさせたいときの処理に適する。
```

```
#include (stdio.h)
int main(void)
{
    int i; //ループ用変数
    for(i = 0; i < 10;i++)
    {
        printf("%d¥n", i);
    }
    return 0;
}
```

```
· while 文
書き方:
while (条件式)
  //処理
動作:
条件式が正しい間ループを行う。
```

for 文の条件判定(継続条件)のみを行うともいえる。 ただし for 文とは違い純粋に条件の判定のみを行うの で、カウントの増加(右のサンプルでは 変数 i)は自分 で行わないといけない。

そのため、実際にはカウントが要らない処理に適す る。

#include <stdio.h></stdio.h>		
int main(void)		
{		
int i = 0; //ループ用変数		
while(i < 10) //i<10の間{}の処理を行う		
{		
printf("%d¥n", i);		
i = i + 1; //i++と書いてもOK		
}		
return 0;		
}		

○その他

実際に条件式で用いることの出来るものには以下のようなものがある。

数学での記号	C言語での記号	備考
\leq	<=	a <= b ⇒ a は b 以下
\geq	>=	a >= b ⇒ a は b 以上
<	<	a < b ⇒ a は b よ り 小さい
>	>	a>b ⇒ aはbより大きい
=	==	$a == b \Rightarrow a \ b \ b \ b \ t$ 等しい
<i>≠</i>	! =	a!=b ⇒ a と b は等しくない
AND	&&	a >= b && a <= c ⇒ a は b 以上かつ c 以下
OR		a >= b a <= c ⇒ a は b 以上または c 以下

· char型

半角文字1文字を格納できる型。 あくまでも1文字であり、文字列はこの ままでは扱えない。 文字列の扱いについては後ほど。

printf で表示させたり scanf で入力を受 け取ったりするには char 型は %c を用

一文字だけを示す場合には' (シングル クォート) で囲む。

例:

printf(" %c ", 'a');

```
#include <stdio.h>
int main(void)
     char moji = 0;
                     //入力された文字を受け取る変数
     printf("なにか文字を入力してください\n->");
     moji = getchar(); //入力された1文字を受け取る
     printf("「%c」が入力されました\n", moji);
     return 0;
```

○演習

- ① 1~入力された値までのすべての整数の合計を表示するプログラムを作れ。 (例: 10 と入力されたら 55 と表示する (1+2+3+・・・+10)
- ②1~入力された値までのすべての2の倍数を表示するプログラムを作れ。

○演習問題の解答例

※解答例のように書かなくても、同じ動作をすれば正解です。

① 1~入力された値までのすべての整数の合計を表示するプログラムを作れ。

(例: 10 と入力されたら 55 と表示する (1+2+3+・・・+10)

```
//入力値までの合計
#include <stdio.h>
int main(void)
      int number; //入力用
                   //計算結果
      int sum;
      int i;
                  //ループ用
      sum = 0; //あとでこれに足していく
      printf("1から入力値までの合計を算出します。\u00a4n");
      printf("値を入力してください。\n-\");
      scanf("%d", &number); //入力
      for (i = 1; i \le number; i++)
             sum += i;
                         //sum = sum + i でもOK
      printf("1から%dまでの合計は%dです。¥n", number, sum);
```

②1~入力された値までのすべての2の倍数を表示するプログラムを作れ。

おまけ

· kbhit()

キーボードのどこかのキーが 押されていたら 1 そうでなければ 0 if 文の条件式に入れるとキー ボードのキーが押されたとき だけ処理を行える

• getch()

押されたキーがどこかを返してくれる。

(char 型)変数に代入すること で調べられる

break

ループ文の中で使うとそのル ープを抜ける処理を行える。

```
#include <stdio.h>
#include <conio.h>
                 //kbhit(), getch()を使うために必要
int main(void)
      char moji = 0; //文字型
      while(1)
                       //条件式は常に1=無限ループ
      {
            if(kbhit()) //キーボードが押されたときに 1 → if文実行
                 moji = getch();//押されたキーを調べる
                 printf(" %c が入力されました。\u20a4n", moji);
                  //↑入力された値の表示
                  if(moji == 'q') //Qキーが押されたら終了
                       printf("終了します\n");
                                   //ループを抜ける
                       break:
                 }
           }
      return 0;
```