

Cプログラミング 入門

—C言語の初歩 (5) : 繰り返し (2) —

早稲田大学

今回の目標

- 繰り返し文が適切に使える。
 - while, do while
 - break, continue
 - 多重ループ

for 文使用プログラム例

1 から 100 までの合計を求める

```
#include <stdio.h>
int main(void){
    int Sum = 0, i;
    printf("Sum=%d¥n",Sum);          /*Sum に入っている値を出力*/
    for (i=1; i<=100; i=i+1){
        Sum = Sum+i ;                /*1つ1つ加算する*/
        printf("+[%d] =%d ¥n",i,Sum); /*足されているか確認*/
    }
    printf("Answer is %d.¥n", Sum);
    return 0;
}
```

while, do while 文

while 文の使い方

```
while (条件式) {  
    繰り返し実行させたいこと  
}
```

- 条件式がチェックされ、偽なら while 文が終了
- { } 内の命令が実行される
- 条件チェックに戻る

while 文を用いたプログラム例 1

sum4.c

```
#include <stdio.h>
int main(void){
    int Sum =0, i;
    i=1;
    while (i<=100){
        Sum +=i;
        i++;
    }
    printf("Answer is %d.¥n", Sum);
    return 0;
}
```

/* 初期設定が必要 */
/* while 文による繰り返し */
/* 繰り返し式も必要 */

- プログラムを写し，実行せよ．プログラム名は sum4.c とせよ．
- while 文は，for 文の初期設定式，繰り返し式がないもの
次のように書いても 1 から 100 までの合計を求めることができる

while, do while 文

do while 文の使い方

```
do {  
    繰り返し実行させたいこと  
} while(条件式);
```

- { } 内の命令が実行される
- 条件式がチェックされ、偽なら do while 文を終了
- { } 内の命令の実行に戻る

do while 文を用いたプログラム例 1

sum5.c

```
#include <stdio.h>
int main(void){
    int Sum =0, i=1;
    do {
        Sum +=i;           /* 1つ1つ加算する */
        i++;
    } while(i<=100);      /* while() の後の「;」を忘れずに */
    printf("Answer is %d.¥n", Sum);
    return 0;
}
```

- do while 文は，条件式が最初はチェックされない
- do の後の { } の中身が，最初は無条件で実行される
- while の後の () 内の条件式が真なら { } 内が繰り返し実行される．
- 最初の 1 回は条件が必ず真と分かっている場合

例題 1

課題：以下のプログラムを作れ．

下記のプログラムを while 文を用いて書け． S_N を以下で定義する：

$$S_N = 4 * \sum_{k=0}^N \frac{(-1)^k}{2k+1}$$

自然数 N を入力し， S_N を計算するプログラムを作れ．

- 表示は小数第 6 位まで出力せよ．
- ファイル名は “piwhile.c” とすること．
- 表示は以下のようにする：

Input positive integer:1000【Enter】

Sum is 3.14?????

break

プログラム例

```
for (i=1; i<=100; i++){  
    if(i==5) break;  
    printf("%d,",i);  
}
```

- ループまたは switch 文から脱出する文
- ループの中で, **break** を実行すると, そのループを強制的に終了する
- 多重ループの場合は, break 文が存在するループのみを終了する
- このプログラムでは以下のように表示される
1,2,3,4,

continue

プログラム例

```
for (i=1; i<=100; i++){  
    if(i==2) continue;  
    if(i==5) break;  
    printf("%d,",i);  
}
```

- ループにおいて、その回のみスキップさせる文
- ループの中で、`continue` を実行すると、そのループを繰り返す { } 内の残りの命令を実行せず、繰り返し式にジャンプする
- このプログラムでは以下のように表示される
1,3,4,

例題 2

課題：以下のプログラムを作れ．

S_N を以下で定義する：

$$S_N = 4 * \sum_{k=0}^N \frac{(-1)^k}{2k+1}$$

誤差 $error$ を入力し， $|S_N - 3.1415| < error$ となるような最小の N を表示するプログラムを作れ．

- ファイル名は"piwhile2.c" とすること．
- 表示は以下のようにする：

Input error :0.01 【Enter】

N is 99.

例題 2 のヒント

- piwhile.c をコピーして編集すること
- 条件が満たされるまでループを回すには無限ループを用いるのが一般的である．無限ループは例えば，以下のようにして使うこと：

```
while(1){  
    やりたいことをかく．  
}
```

- $|S_N - 3.1415| < \text{error}$ は

$$-\text{error} < S_N - 3.1415 < \text{error}$$

と同じ意味であることに注意せよ．

- 最小の N を表示するには最終的に何を表示させればよいか？

例題 3

課題：以下のプログラムを作れ

- 九九の表の中身を計算して、表示させるプログラムを作れ
- 表示は以下のようにする

1	2	3	4	5	6	7	8	9
2	4	6	8	10	12	14	16	18
3	6	9	12	15	18	21	24	27
4	8	12	16	20	24	28	32	36
5	10	15	20	25	30	35	40	45
6	12	18	24	30	36	42	48	54
7	14	21	28	35	42	49	56	63
8	16	24	32	40	48	56	64	72
9	18	27	36	45	54	63	72	81

例題3のヒント

- **二重ループ**を使う．変数は2つ必要になる．

```
int i, j;
for (i=1;...;...){
    for(j=1;...;...){
        ....
    }
    ....
}
```

- 整数の桁を揃えて表示するには，printfで"%3d"
- はじめは1行目を出力するプログラムを作り，それができたら2行目を出力するプログラムを作ろう

1 2 3 4 5 6 7 8 9

まとめ

- 繰り返し文
 - while, do while
 - break, continue
 - 多重ループ