

繰り返し操作の表現

1. 繰り返し操作の記述

繰り返し操作は（人間との比較において）コンピュータが非常に有利な作業です。while, for, do whileの3つの表現形式の記述と違いに慣れて下さい。

2. 具体的な記述形式

< for文の基本形 >

```
for(i=1; i<=10; i++){
    printf("%d\n", i);
}
```

for(初期値; 条件; 更新)

< while文の基本形 >

```
i=1;
while(i<=10){
    printf("%d\n", i);
    i++;
}
```

while(条件)

上記の表現が繰り返し操作記述の基本形です。例えば、iを1ずつ増やしながらprintfを10回繰り返すプログラムを記述すると以下ようになります。

```
#include <stdio.h>

int main(void)
{
    int i;
    for(i=1; i<=10; i++){
        printf("%d\n", i);
    }
}
```

```
#include <stdio.h>

int main(void)
{
    int i;
    i=1;
    while(i<=10){
        printf("%d\n", i);
        i++;
    }
}
```

どちらも繰り返しループの入口で条件を調べ、設定した条件を満足する範囲で以下を実行することになります。これに対し、do while文では、

```
i=1;
do{
    printf("%d\n", i);
    i++;
}while(i<=10);
```

のように、ループの後で条件について調べるので、最初の1回は必ず実行されることになります。

パラメータを少しずつ増減していく操作の記述に簡略的な表記法が使える点がCの特徴です。

count=count+1 -> count++
count=count-1 -> count--

odd=odd+2 -> odd+=2
odd=odd-2 -> odd-=2

3. 入力データの数が決まっていない場合の対応方法

while(scanf("%f", &x)==1)

何か値が入力されている限りループ

while(scanf("%f", &x)!=EOF)

EOF（終端コード）でない限りループ

課題

- (1) テキスト15ページ図14のプログラムで、whileループの中にprintf文を入れて、x, count, oddがどのように変化していくか確認して下さい。
- (2) テキスト17ページに「1つの式に2度以上現れる変数に対してインクリメント演算子やデクリメント演算子を使うのは好ましくない」という記述がありますが、実際に $n*n++$ を設定するとどのようになるか確かめて下さい。
- (3) キーボードから10を入力すると次のように出力するプログラムを、while, for, do while文のそれぞれを使って書いて下さい。

```
1
2
3
4
5
6
7
8
9
10
```

- (4) キーボードから10を入力すると次のように出力するプログラムを書いて下さい。

```
1 2 3 4 5 6 7 8 9 10
2 3 4 5 6 7 8 9
3 4 5 6 7 8
4 5 6 7
5 6
```

- (5) while文を使って、 $1^2+2^2+3^2+\dots+n^2 < 1,000,000$ を満たす最大の整数nを求めるプログラムを書いて下さい。
- (6) 任意の3つの整数を（キーボード入力で）読み込んで、平均と標準偏差を小数点以下1桁で表示するプログラムを書いて下さい。
- (7) 任意の3つの整数を（キーボード入力で）読み込んで、最大値と最小値を表示するプログラムを書いて下さい。
- (8) キーボードから整数を入力し、その数だけ*を表示するプログラムを書いて下さい。
- (9) キーボードから整数を入力し、その段数のピラミッドを*を使って表示するプログラムを書いて下さい。

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

- (10) 正の整数を1つ入力して、0になるまでカウントダウンしながら、その数だけ*を表示するプログラムを書いて下さい。次に、sleep文（`#include <unistd.h>`が必要）を使って、1秒に1つカウントダウンされるようにして下さい。さらに、エスケープシーケンスを利用して1行に表示が上書きされるようにし、*の列が徐々に短くなっていくようにして下さい。注：fflush(stdout)を使う（おまじないで最後につけて下さい）