

Lecture 7

プログラミング演習 I

本日の演習の流れ

1. 論理値の演算
2. 入れ子構造
3. 再帰関数の利用

course page

<http://amth.mind.meiji.ac.jp/courses/PE1/>

論理値の演算

演算	意味	結果
$x \parallel y$	OR (論理和)	xとyのどちらかが真なら真
$x \&\& y$	AND (論理積)	xとyの両方が真なら真
$!x$	NOT (否定)	xが偽なら真, xが真なら偽

1

整数型変数 x, y において, y の値が 10 以下もしくは 20 以上のときは x に 100 を, それ以外は x に 200 を代入する if 文を書け.

ヒント :

2つの条件のどちらかが満足されるとき, という式で表すには論理演算子を用いる. この場合は, 論理和 (or) を使う.

||

論理値の演算

2 整数型変数 x が $0 < x < 5$ であるかどうかを判断するのに

```
if ( 0 < x < 5 ) {  
    文 ;  
}
```

と書くことができるか.

論理値の演算

- 3 次の連立方程式において、係数の a, b, c, d, e, f を入力すると解 x, y が表示されるプログラムを書け.

$$ax + cy = e$$

$$bx + dy = f$$

ただし、 x, y は次の式で与えられるとする.

$$x = \frac{ed - cf}{ad - bc}, \quad y = \frac{af - be}{ad - bc}$$

注意：

1. $ad - bc = 0$...

論理値の演算

回答

```
1  #include <stdio.h>
2  int main(void){
3      double a,b,c,d,e,f;
4      double x,y;
5      double det;
6      printf("a=? ");   scanf("%lf",&a);
7      printf("b=? ");   scanf("%lf",&b);
8      printf("c=? ");   scanf("%lf",&c);
9      printf("d=? ");   scanf("%lf",&d);
10     printf("e=? ");   scanf("%lf",&e);
11     printf("f=? ");   scanf("%lf",&f);
12     det = a*d-b*c;
13     if(det != 0){
14         x = (e*d-c*f)/det; y = (a*f - b*e)/det;
15         printf("x=%lf, y=%lf \n",x,y);
16     }else{
17         printf("Unique solution does not exist..\n");
18     }
19     return 0;
20 }
```

```
a=? 1.0
b=? 2.0
c=? 3.0
d=? 4.0
e=? 5.0
f=? 6.0
x=-1.000000, y=2.000000
```

入れ子構造 (nesting)

if文やif～else分は基本的に 2 者択一の分岐を行うものである。しかし、if分の中にもう 1 つのif文を付け加えて、複雑な分岐をさせることができる (入れ子構造という)。

- 4 if文はネストされることが可能である。次の例では、`a == b` が偽 (false) となった場合に実行される `else` 文の直後のブロック中に、別の if 文が入っている。

```
1 #include <stdio.h>
2 int main(void){
3     int a = 3;
4     int b = 3;
5     if(a == b){
6         printf("a is equal to b\n");
7     }else{
8         if(a < b){
9             printf("a is smaller than b\n");
10        }else{
11            printf("a is greater than b\n");
12        }
13    }
14    return 0;
15 }
```

NOTE:

if文に限らず、プログラムの様々な要素についてネスト構造を使うことが可能である。

for文の復習

- 5 次の計算を求め、表示するプログラムをfor文を使い書きなさい.

$$wa = 5^3 + 6^3 + 7^3 + 8^3 + 9^3$$

ヒント： 実行結果：1925

while 文の復習

6

while分を用いて, x^n (n は負でない整数) の値を計算するプログラムを作成せよ.

for文の入れ子構造

- 7 ディスプレイ上に右のように表示するプログラムを for文を使って書け.

```
Lec6_Pro_2019 ./a.out
```

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

ヒント :

for文の入れ子構造 (2重ループや2重for文などと呼ぶ) を使用すること.

ヒント :

```
for(){  
    for (){  
  
    }  
}
```

while文による無限ループ (infinite loop)

8

つぎのような繰り返し動作をするプログラムを作成せよ.

- (1) ユーザーから数値の代入を受ける
- (2) 入力された数値が奇数なら odd を表示し, (1) に戻り繰り返す
- (3) 入力された数値が偶数なら even を表示し, (1) に戻り繰り返す
- (4) 入力された数値がマイナスなら, 何も表示せず終了する

ヒント : 無限ループと除算の余りを計算する演算子「%」を使用する.

```
printf("even: %d\n", 8%2);  
printf("odd: %d\n", 9%2);
```

再帰版 (recursion)

自分自身を呼び出す関数は再帰関数 (recursive function) と呼ぶ。

9 再帰版のfact(n)を書いてみよう.

n!

ヒント :	fact(1)=1		fact(1)=1
	fact(2)=2 x 1		fact(2)=2 x fact(1)
	fact(3)=3 x 2 x 1	→	fact(3)=3 x fact(2)
	fact(4)=4 x 3 x 2 x 1		fact(4)=4 x fact(3)
	fact(5)= ...		fact(5)= ...

右辺の式を関数 fact(n) で置き換えられると気づきませんか？

$$\begin{cases} fact(1) = 1 \\ fact(n) = n \times fact(n - 1) \quad (n > 1) \end{cases}$$