

CP 111 Computer Programming, 2/2554, Section B01/B02

Programming Homework 03

กำหนดส่งภายในวันจันทร์ที่ 9 ม.ค. 2555 เวลา 23.59 น.

คำเตือน

นิสิตสามารถจับกลุ่มปรึกษาหารือกันได้ แต่นิสิตแต่ละคนในกลุ่มจะต้องลงมือเขียนโปรแกรมด้วยตนเองและส่งโปรแกรมของตนเองโดยไม่ขึ้นอยู่กับสมาชิกในกลุ่ม และอย่านำโปรแกรมของตนเองให้กับผู้อื่นดู
ถ้าตรวจพบว่านิสิตนำงานของคนอื่นมาส่งหรือนิสิตก๊อปปี้ แก้ไข ดัดแปลงโปรแกรมของผู้อื่น จะมีโทษคือปรับตก (ได้เกรด E) ในวิชานี้ทันที

หมายเหตุ

1. ข้อกำหนดการตั้งชื่อไฟล์การบ้านคือ xxx_HWyy_Qzz.* โดยที่ xxx = รหัสสามตัวท้ายของเลขประจำตัว, yy = หมายเลข homework, zz = หมายเลขข้อในแต่ละ homework (ตัวอย่างเช่น ไฟล์โปรแกรม ข้อ 1 การบ้านครั้งที่ 1 ของนิสิตรหัส 000 คือ 000_HW01_Q01.c) และในแต่ละข้อ
2. การส่งการบ้าน ให้ส่งไฟล์โปรแกรมไปตรวจกับระบบ Café-grader ที่ <http://cslab.swu.ac.th/grader>
3. การเขียนโปรแกรมที่คำนวณกับตัวเลขที่มีจุดทศนิยม ให้ประกาศตัวแปรที่เกี่ยวข้องเป็นแบบ double (Double-precision floating point)

1. Problem name: Straight_line

จงเขียนโปรแกรมเพื่อรับค่าตัวเลขจำนวนเต็มทั้งหมด 6 ค่า โดยให้รับบรรทัดละ 2 ค่าทั้งหมด 3 บรรทัด โดยกำหนดค่าที่อ่านเข้ามาคือค่า $x_1, y_1, x_2, y_2, x_3, y_3$ ในรูปแบบต่อไปนี้

$x_1 \ y_1$

$x_2 \ y_2$

$x_3 \ y_3$

โดยกำหนดให้ $x_i, y_i \ i=1,2,3$ แทนค่า xy-coordinate ของจุดหนึ่งๆ ในระนาบ 2 มิติ หลังจากนั้นให้ตรวจสอบว่าจุดทั้งสามนี้อยู่บนเส้นตรงเดียวกันหรือไม่ ถ้าอยู่บนเส้นตรงเดียวกันให้แสดงข้อความ YES มิฉะนั้นให้แสดงข้อความ NO ในบรรทัดถัดไป

ตัวอย่างการทำงานของโปรแกรม

ข้อมูลนำเข้า (Input)	ข้อมูลส่งออก (Output)
15 15 10 10 -40 -40	YES
15 -15 -10 10 -40 -40	NO
10 10 -20 10 0 10	YES
0 -12 0 12 -12 0	NO

2. Problem name: Perfect_number

นิยาม Perfect Number คือ จำนวนเต็มบวกที่ค่าของมันมีค่าเท่ากับผลรวมของตัวเลขที่หารมันลงตัว (ยกเว้นตัวมัน) เช่น 6 เป็น Perfect number เพราะ 6 หารด้วย 1, 2, 3 ลงตัวและ $6=1+2+3$ หรือ 28 เป็น Perfect number เพราะ 28 หารด้วย 1, 2, 4, 7, 14 ลงตัวและ $28=1+2+4+7+14$

จงเขียนโปรแกรมตรวจสอบว่าจำนวนที่ได้รับเข้ามาเป็น perfect number หรือไม่

Input	Output
จำนวนเต็มบวก N	ถ้า N เป็น perfect number ให้พิมพ์ว่า N is a perfect number. ไม่เช่นนั้นให้พิมพ์ N is not a perfect number.

ตัวอย่างการทำงานของโปรแกรม

ข้อมูลนำเข้า (Input)	ข้อมูลส่งออก (Output)
28	28 is a perfect number.
100	100 is not a perfect number.

Hint:

```
#include <stdio.h>
int main()
{
    int i,x,sum;
    scanf("%d",&x);
    sum=
    for(i=1; ... ; ... )
    {
        if(...)
        {
            sum=...;
        }
    }
    if(...)
        printf("%d is a perfect number.\n",x);
    else
        printf("%d is not a perfect number.\n",x);
    return 0;
}
```

3. Problem name: Second_largest

จงเขียนโปรแกรมหาค่ามากที่สุดอันดับสอง (Second largest value) ของข้อมูล M ตัว ($M > 0$) โดยข้อมูลที่ป้อนเข้ามา (input) มีรูปแบบ (format) ดังนี้

1. บรรทัดแรกบอกจำนวนของข้อมูลทั้งหมด สมมุติ M
2. ตั้งแต่บรรทัดที่ 2 ไปเป็นจำนวน M บรรทัดเป็นข้อมูลแต่ละตัว โดยกำหนดให้ข้อมูลเป็นตัวเลขแบบจำนวนเต็ม

เมื่อได้รับ input ครบแล้ว ให้แสดงค่ามากที่สุดอันดับสองออกสู่หน้าจอ

ตัวอย่างการทำงานของโปรแกรม

ข้อมูลนำเข้า (Input)	ข้อมูลส่งออก (Output)
5 881 325 732 -520 983	881
4 -2345678 -1345678 -1234567 -1457890	-1345678
6 2 1 1 4 2 5	4
2 -100 -10	-100

Hint: ดู Paper homework 3

4. Problem name: LCM

นิยาม ตัวคูณร่วมน้อย หรือ ค.ร.น. (Least Common Multiple or LCM) ของตัวเลข 2 จำนวนใดๆ คือจำนวนที่น้อยที่สุดที่ตัวเลขทั้งสองสามารถหารจำนวนนี้ลงตัว

จงเขียนโปรแกรมทำการรับค่าตัวเลขจำนวนเต็มบวกที่ไม่เท่ากับศูนย์ 2 จำนวนจากผู้ใช้ หลังจากนั้นให้ทำการหาตัวคูณร่วมน้อย หรือ ค.ร.น. (Least Common Multiple or LCM) ของ 2 จำนวนนี้ แล้วนำมาแสดงผลออกสู่หน้าจอ

ตัวอย่างการทำงานของโปรแกรม

ข้อมูลนำเข้า (Input)	ข้อมูลส่งออก (Output)
2 3	6
105 201	7035
123 456	18696

Hint

```
#include <stdio.h>

int main()
{
    int x,y,lcm_val,i,max;

    scanf("%d %d",&x,&y);
    max=
    for(i=max;...;...)
    {
        if( )
        {
            lcm_val=
        }
    }
    printf("%d\n",lcm_val);
    return 0;
}
```

5. Problem name: Numeric_Triangle_2

จงเขียนโปรแกรมที่รับ input เป็นค่าจำนวนเต็มบวก สมมติว่ามีค่าเท่ากับ N หลังจากนั้นแสดงรูปแบบดังตัวอย่างการทำงานของโปรแกรม นั่นคือ

ในบรรทัดที่ 1 ให้แสดงตัวเลข 1

ในบรรทัดที่ 2 ให้แสดงตัวเลข 2 3

ในบรรทัดที่ 3 ให้แสดงตัวเลข 4 5 6

ในบรรทัดที่ 4 ให้แสดงตัวเลข 7 8 9 10

ในบรรทัดที่ 4 ให้แสดงตัวเลข 11 12 13 14

...

ในบรรทัดที่ N-1 ให้แสดงตัวเลขทั้งหมด N-2 ตัวที่ค่าเรียงต่อกันจากบรรทัดที่ N-2

ในบรรทัดที่ N ให้แสดงตัวเลขทั้งหมด N ตัวที่ค่าเรียงต่อกันจากบรรทัดที่ N-1

ตัวอย่างการทำงานของโปรแกรม

ข้อมูลนำเข้า (Input)	ข้อมูลส่งออก (Output)
5	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
7	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28

Hint: ดูโจทย์ Numeric_Triangle

6. Problem name: Duplicated_entry

จงเขียนโปรแกรมรับข้อมูลแบบจำนวนเต็มที่มีค่าอยู่ในช่วง 1-15 ทั้งหมด 16 จำนวน ซึ่งจะถูกป้อนผ่านคีย์บอร์ดเข้ามาทีละบรรทัด โดยกำหนดให้ทั้ง 16 จำนวนนี้จะต้องมีตัวเลขทั้ง 15 ค่า ดังนั้นจะต้องมีค่าตัวเลขที่ปรากฏอยู่สองครั้งในข้อมูลที่ป้อนเข้ามา จงหาตัวเลขที่ซ้ำนั้นแล้วแสดงออกสู่หน้าจอ

ตัวอย่างการทำงานของโปรแกรม

ข้อมูลนำเข้า (Input)	ข้อมูลส่งออก (Output)
5 9 1 10 7 2 8 12 15 11 3 4 13 9 14 6	9

Hint:

ถ้ากำหนดให้ $\text{sum} = 1+2+3+4+5+\dots+15$ และสมมติว่า x คือค่าตัวเลขที่ซ้ำ

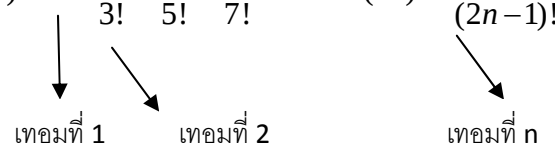
ดังนั้นผลรวมของข้อมูลอินพุตทั้ง 16 ตัวคือ $1+2+3+4+5+\dots+15+x$ จะมีค่าเท่ากับ $\text{sum} + x$

ดังนั้น $x = ?$

7. Problem name: sine_approx

จงเขียนโปรแกรมที่ทำการคำนวณค่าประมาณของ $\sin(x)$ โดยที่ x มีหน่วยเป็น radians โดยใช้สูตร

$$\sin(x) \approx x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots + (-1)^{n-1} \frac{x^{(2n-1)}}{(2n-1)!} = \sum_{k=1}^n (-1)^{k-1} \frac{x^{(2k-1)}}{(2k-1)!}$$



กำหนดให้

π radians=180 degree และ $\pi = 3.1416$

โปรแกรมจะรับอินพุต 2 จำนวนคั่นด้วยช่องว่างในบรรทัดเดียวกัน โดยอินพุตแรกคือค่า x มีหน่วยเป็น degree ส่วนอินพุตตัวที่สองคือค่า n ซึ่งจะคือจำนวนเทอมที่จะใช้ในการคำนวณค่าประมาณของ $\sin(x)$ ตามสูตรข้างบน

หลังจากนั้นโปรแกรมจะต้องแสดงค่าประมาณ $\sin(x)$ ที่คำนวณได้ออกสู่หน้าจอโดยให้แสดงเลขทศนิยม 6 ตำแหน่ง

Note: x ในสูตรการประมาณค่า sine มีหน่วยเป็น radians แต่โปรแกรมจะรับค่าอินพุตเป็นหน่วย degree ดังนั้นจึงต้องแปลงค่าที่รับมาจากอินพุตให้เป็นอยู่ในรูปของหน่วย radians ก่อนนำไปคำนวณ

ตัวอย่างการทำงานของโปรแกรม

ข้อมูลนำเข้า (Input)	ข้อมูลส่งออก (Output)
90 20	1.000000
90 3	1.004525
45 20	0.707108
30 100	0.500001
60 50	0.866027
0 10	0.000000
15 50	0.258820
85 100	0.996195
15 20	0.258820
24 20	0.406738

Hint:

```
#include <stdio.h>

int main()
{
    double sin_x,x,xp,fact;
    int k,n;

    scanf("%lf %d",&x,&n);

    x=x*(3.1416/.....); /* แปลง x จากหน่วย degrees ให้เป็น radians */

    xp=x;    /* xp เก็บค่า x -> x3 -> x5 -> x7 -> ... */

    fact=1; /* fact เก็บค่า 1! -> 3! -> 5! -> 7! -> ... */

    sin_x =.....;

    for(k=1;k<=n;k++ )
    {
        if(k%2==.....)
            sin_x=sin_x+.....
        else
            sin_x=sin_x-.....

        /* update ค่า xp และ fact สำหรับใช้ใน iteration ถัดไป */

        xp=xp*.....;

        fact=fact*.....;
    }

    printf("%.6f\n",sin_x);

    return 0;
}
```

8. Problem name: sum_n

จงเขียนโปรแกรมเพื่อทำการหาผลรวมของจำนวนเต็มที่มีค่า $i = nstart, nstart+n, nstart+2n, \dots, nstart+3n, \dots$ ไปเรื่อยจนกว่า i จะมีค่ามากกว่า $nstop$ โดยที่ค่า $nstart, nstop$ และ n จะถูกป้อนเข้ามาทางคีย์บอร์ด เช่น ถ้าผู้ใช้ป้อน $nstart=2, nstop=100, n=3$ โปรแกรมจะคำนวณหาค่า $2+5+8+11+\dots+95+98 (=1650)$

ตัวอย่างการทำงานของโปรแกรม

ข้อมูลนำเข้า (Input)	ข้อมูลส่งออก (Output)
1 10 1	55
2 100 3	1650
123 10000 12	4170264
1 1000 10	49600
-14 -2 2	-56
-14 10 1	-50

Hint: ดู Paper homework 3

9. Problem name: fibo_mai_pee (ไฟบอยี่พี) นำมาจาก codejom

สำหรับจำนวนเต็ม m และ n สองจำนวน ลำดับไฟบอยี่พี ที่มีพารามิเตอร์ (m, n) เป็นลำดับที่มีคุณสมบัติดังนี้

- $f(0) = m$
- $f(1) = n$
- $f(i) = f(i-1) + f(i-2)$ เมื่อ $\text{floor}(i/2) \bmod 2 = 1$
- $f(i) = f(i-1) - f(i-4)$ เมื่อ $\text{floor}(i/2) \bmod 2 = 0$

โดยที่ $f(i)$ คือสมาชิกตัวที่ i ของลำดับ และ $\text{floor}(x)$ คือจำนวนเต็มที่เกิดจากการปัดเศษทศนิยมลงของจำนวน x ตัวอย่างเช่น ลำดับไฟบอยี่พี ที่มีพารามิเตอร์ $(2, 3)$ มีส่วนต้นของลำดับดังนี้

$f(0) = 2$
 $f(1) = 3$
 $f(2) = 3 + 2 = 5$ ($\text{floor}(2/2) \bmod 2 = 1 \bmod 2 = 1$)
 $f(3) = 5 + 3 = 8$ ($\text{floor}(3/2) \bmod 2 = 1 \bmod 2 = 1$)
 $f(4) = 8 - 2 = 6$ ($\text{floor}(4/2) \bmod 2 = 2 \bmod 2 = 0$)
 $f(5) = 6 - 3 = 3$ ($\text{floor}(5/2) \bmod 2 = 2 \bmod 2 = 0$)
 $f(6) = 3 + 6 = 9$ ($\text{floor}(6/2) \bmod 2 = 3 \bmod 2 = 1$)
 $f(7) = 9 + 3 = 12$ ($\text{floor}(7/2) \bmod 2 = 3 \bmod 2 = 1$)
 $f(8) = 12 - 6 = 6$ ($\text{floor}(8/2) \bmod 2 = 4 \bmod 2 = 0$)
 $f(9) = 6 - 3 = 3$ ($\text{floor}(9/2) \bmod 2 = 4 \bmod 2 = 0$)

...

ให้หาค่าของสมาชิกตัวที่ i จากลำดับไฟบอยี่พี ที่มีพารามิเตอร์ (m, n) ที่กำหนดให้

ข้อมูลนำเข้า

จำนวนเต็มสามจำนวน m, n, i เว้นแต่ละตัวด้วยช่องว่างหนึ่งช่อง ($0 \leq m, n \leq 30$ และ $0 \leq i < 1,000$)

ข้อมูลส่งออก

สมาชิกตัวที่ i ในลำดับไฟบอยี่พี ที่มีพารามิเตอร์ (m, m)

ตัวอย่างการทำงานของโปรแกรม

ข้อมูลนำเข้า (Input)	ข้อมูลส่งออก (Output)
2 3 6	9
2 3 9	3
5 3 5	3
5 3 7	12
2 3 0	2
2 3 1	3

```

#include <stdio.h>
int main()
{
    int m,n,i;
    int k,f_k,f_k_1,f_k_2,f_k_3,f_k_4;

    scanf("%d %d %d",&m,&n,&i);
    f_k_1=    /* f(k-1) */
    f_k_2=    /* f(k-2) */
    f_k_3=0; /* f(k-3): 0 or any value is okay for initialization. */
    f_k_4=0; /* f(k-4): 0 or any value is okay for initialization. */
    if( )
        f_k=
    else if( )
        f_k=
    else
    {
        for(k=2; ; )
        {
            if( )
                f_k=
            else
                f_k=

            /* update f_k_4,f_k_3,f_k_2,f_k_1 */

        }
    }
    printf("%d\n",f_k);
    return 0;
}

```