

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

Programming Homework 02

กำหนดส่งภายในวันอังคารที่ 3 ม.ค. 2554 เวลา 23.59 น.

คำเตือน

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

หมายเหตุ

- 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
- การส่งการบ้าน ให้ส่งไฟล์โปรแกรมไปตรวจกับระบบ Café-grader ที่ <http://cslab.swu.ac.th/grader>
- การเขียนโปรแกรมที่คำนวณกับตัวเลขที่มีจุดทศนิยม ให้ประกาศตัวแปรที่เกี่ยวข้องเป็นแบบ double (Double-precision floating point)

1. Problem name: Range_of_Data

จงเขียนโปรแกรมเพื่อคำนวณหาพิสัย (Range) ของข้อมูลจำนวนเต็มบวก โดยข้อมูลนำเข้าจะเป็นจำนวนเต็มบวกบรรทัดละตัว และสิ้นสุดด้วย -1 เมื่อสิ้นสุดข้อมูลนำเข้าให้แสดงค่าพิสัยข้อมูลชุดนั้น

นิยาม พิสัย (Range) ของข้อมูลกลุ่มหนึ่งๆ มีค่าเท่ากับ ค่ามากที่สุด (Maximum value) ลบด้วยค่าน้อยสุด (Minimum value)

ตัวอย่างการทำงานของโปรแกรม

ข้อมูลนำเข้า (Input)	ข้อมูลส่งออก (Output)
12 31 10 1 41 32 -1	40

Note: จากตัวอย่างชุดทดสอบ (Test case) ข้างบน ค่า max คือ 41 ค่า min คือ 1 ดังนั้น พิสัย (range) มีค่าเท่ากับ $41-1=40$

2. Problem name: Div_5_7_11

จงเขียนโปรแกรมพิมพ์ค่าตัวเลขตั้งแต่ 1 ถึง N ที่หารด้วย 5 หรือ 7 ลงตัว แต่หารด้วย 11 แล้วเหลือเศษ 3 โดยให้พิมพ์ที่ละบรรทัดเรียงจากน้อยไปมาก กำหนดให้ค่า N ถูกป้อนจากคีย์บอร์ดโดยผู้ใช้

ตัวอย่างการทำงานของโปรแกรม

ข้อมูลนำเข้า (Input)	ข้อมูลส่งออก (Output)
200↵	14 25 80 91 135 168 190

Hint:

```
#include <stdio.h>

int main()
{
    int i,N;

    scanf("%d",&N);

    for(i=1;...;...)
    {
        if(...)
        {
            printf("%d\n",i);
        }
    }
    return 0;
}
```

3. Problem name: Mean_Min_Max_2

จงเขียนโปรแกรมหาค่าเฉลี่ย (Mean) ค่าน้อยสุด (Minimum value) และค่ามากที่สุด (Maximum value) ของข้อมูลแบบจำนวนเต็มที่ไม่ใช่ค่าลบ (non-negative integers) โดยข้อมูลจะถูกป้อนเข้ามาที่ละบรรทัดไปเรื่อย ๆ จนกว่าจะเจอข้อมูลที่เป็นจำนวนเต็มลบ (เช่น -1 หรือ -2 หรือ -3 เป็นต้น) ซึ่งจะถือว่าเป็นจุดสิ้นสุดของข้อมูล (ค่าจำนวนเต็มลบนี้จะไม่นับรวมอยู่ในชุดข้อมูลที่จะคำนวณ Mean, Min, Max) เมื่อได้รับ input ครบแล้ว ให้แสดงค่าเฉลี่ย (แสดงเลขทศนิยม 2 หลัก) ค่าน้อยที่สุด และค่ามากที่สุด ออกมาที่หน้าจอที่ละบรรทัด

ตัวอย่างการทำงานของโปรแกรม

ข้อมูลนำเข้า (Input)	ข้อมูลส่งออก (Output)
6↵ 5↵ 0↵ 1↵ 2↵ 3↵ -1↵	2.83 0 6
40↵ 30↵ 40↵ 50↵ -100↵	40.00 30 50

ดู Hint หน้าถัดไป สำหรับ Mean_Min_Max_2:

Hint สำหรับ Mean_Min_Max_2:

```
#include <stdio.h>
int main()
{
    int N, sum, min, max, x;
    double mean;

    N=...;
    sum=...;
    do
    {
        scanf("%d", &x);
        if(x>=...)
        {
            // Update min max
            if(N==...)
            {
                min=x;
                max=x;
            }
            else
            {
                // Update sum N
                N=...;
                sum=...;
            }
        }
    } while(x>=...);

    // calculate the mean
    mean=...;

    // display the results
    printf(    );
    printf(    );
    printf(    );

    return 0;
}
```

4. Problem name: Div_by_k

จงเขียนโปรแกรมเพื่อรับค่าที่มีชนิดข้อมูลแบบจำนวนเต็ม 3 จำนวน สมมติว่าเป็น k, m และ n ตามลำดับ โดยที่กำหนดว่า $k > 1$ และ $m < n$ หลังจากนั้นให้ทำการหาค่าในช่วง m และ n (รวมทั้ง m และ n) ค่าใดที่สามารถหารด้วย k ลงตัว แล้วแสดงค่าเหล่านั้นออกสู่หน้าจอที่ละบรรทัด นอกจากนี้ให้ทำการหาผลรวมของค่าตัวเลขเหล่านั้นและแสดงที่บรรทัดสุดท้าย

ตัวอย่างการทำงานของโปรแกรม

ข้อมูลนำเข้า (Input)	ข้อมูลส่งออก (Output)
3 10 20↵	12 15 18 45

หมายเหตุ ในชุดทดสอบ (Test case) ข้างบน Input k=3, m=10, n=20 เราพบว่าในช่วง m ถึง n (10 ถึง 20) ตัวเลข 12 15 18 คือ ค่าที่อยู่ในช่วง 10 ถึง 20 ที่สามารถหารด้วย k (k=3) ลงตัว และ ผลรวมของค่าเหล่านี้คือ $12+15+18=45$

```
#include <stdio.h>
int main()
{
    int i,k,m,n,sum;

    scanf("%d %d %d",&k,&m,&n);
    sum=0;
    for(i= ;i<= ; )
    {
        if( )
        {
            printf("%d\n",i);
            sum= ;
        }
    }
    printf( );

    return 0;
}
```

5. Problem name: GCD

จงเขียนโปรแกรม ทำการรับค่าตัวเลขจำนวนเต็มบวก 2 ตัวจากผู้ใช้ หลังจากนั้นให้ทำการหาค่าตัวหารร่วมมาก หรือ ห.ร.ม. (Greatest Common Divisor or GCD) ของจำนวนเต็มบวก 2 จำนวน แล้วนำมาแสดงผลออกสู่หน้าจอ

ตัวอย่างการทำงานของโปรแกรม

ข้อมูลนำเข้า (Input)	ข้อมูลส่งออก (Output)
49 84↵	7
125 75↵	25

Hint: วิธีแบบง่าย ๆ ของการหาค่าตัวหารร่วมมากของจำนวนเต็มบวก 2 จำนวน คือ ตรวจสอบว่าตั้งแต่ ค่า 1, 2, 3,... จนถึง ค่าของตัวเลขใดตัวเลขหนึ่งใน 2 ตัวนี้ (ตัวที่ต้องการหา ห.ร.ม.) ค่าใดคือค่าที่มากที่สุดที่สามารถหารตัวเลขทั้งสองจำนวนลงตัว ซึ่งค่าตัวเลขนั่นก็คือค่าตัวหารร่วมมาก ตัวอย่างเช่น ค่าตัวหารร่วมมากของ 6 และ 12 คือ 6 เพราะว่า ตั้งแต่ 1, 2, 3, 4,... ,12 เราพบว่าค่า 1, 2, 3 และ 6 สามารถหาร 6 และ 12 ลงตัว แต่ 6 คือค่าที่มากที่สุดในบรรดาค่าเหล่านี้

```
#include <stdio.h>
int main()
{
    int i,x1,x2,tmp,gcd;

    scanf("%d %d",&x1,&x2);

    for(i=1;i<= ;    )
    {
        if(x1%i==0 &&    ){
            gcd=
        }
    }
    printf("%d\n",gcd);

    return 0;
}
```

6. Problem name: Deposit_Interest_loop

จงเขียนโปรแกรมเพื่อคำนวณเงินทั้งหมดที่ได้จากการฝากเงินกับธนาคารแห่งหนึ่ง โดยให้แสดงเงินเมื่อสิ้นสุดในแต่ละปีเป็นระยะเวลา N ปี โดยกำหนดให้โปรแกรมรับข้อมูลนำเข้าจากผู้ใ้ 3 ตัวคือค่าจำนวนเงินที่ต้องการจะฝาก (สมมติว่าเป็นหน่วยบาท) อัตราดอกเบี้ยในรูปของเปอร์เซ็นต์ (0-100) และจำนวนปีที่ต้องการแสดง (N) หลังจากนั้นโปรแกรมจะต้องแสดงผลลัพธ์จำนวนเงินสุทธิในแต่ละปีเป็นเลขทศนิยม 2 หลักที่ละบรรทัด (จำนวนบรรทัดเท่ากับ N)

ตัวอย่างการทำงานของโปรแกรม

ข้อมูลนำเข้า (Input)	ข้อมูลส่งออก (Output)
23451.75 1.8 10↵	23873.88 24303.61 24741.08 25186.42 25639.77 26101.29 26571.11 27049.39 27536.28 28031.93
1000 1.75 7↵	1017.50 1035.31 1053.42 1071.86 1090.62 1109.70 1129.12

ดู Hint หน้าถัดไป สำหรับ Deposit_Interest_loop:

Hint สำหรับ Deposit_Interest_loop:

```
#include <stdio.h>
int main()
{
    double deposit,i_rate,total;
    int i,N;

    scanf("%lf %lf %d",&deposit,&i_rate,&N);
    for(i=1;... ; ...)
    {
        total=deposit+((.../100)*...);
        printf("%.2f\n",total);
        deposit=.....;    /* deposit (เงินฝากที่จะคิดดอกเบี้ยในเดือนถัดไป) เท่ากับ เงินรวมเดือนนี้ */
    }
    return 0;
}
```

7. Problem name: Neg_Pos_Avg

จงเขียนโปรแกรมเพื่อทำการรับค่าเลขจำนวนเต็มที่มีไม่เท่ากับศูนย์ (Nonzero integers) ผ่าน keyboard เข้ามาเรื่อยๆ โดยจะหยุดการรับค่าก็ต่อเมื่อผู้ใช้ป้อนค่า 0 หลังจากนั้นโปรแกรมจะแสดงค่าเฉลี่ยที่คำนวณจากจำนวนบวกทั้งหมด (Average of positive numbers) และค่าเฉลี่ยที่คำนวณจากจำนวนลบทั้งหมด (Average of negative numbers) แล้วแสดงออกสู่หน้าจอที่ละบรรทัด โดยให้แสดงจุดทศนิยม 2 ตำแหน่ง

ตัวอย่างการทำงานของโปรแกรม

ข้อมูลนำเข้า (Input)	ข้อมูลส่งออก (Output)
-1 10 -5 15 -3 -7 5 0	10.00 -4.00
15 -16 17 -18 19 -20 21 22 -23 0	18.80 -19.25

ดู Hint หน้าถัดไป สำหรับ Mean_Min_Max_2:

Hint: Neg_Pos_Avg

```
#include <stdio.h>
int main()
{
    int sum_pos, sum_neg, N_pos, N_neg, data;
    double avg_pos, avg_neg;

    sum_pos=0; /* sum_pos เก็บผลรวมของค่าบวก */
    sum_neg= ; /* sum_neg เก็บผลรวมของค่าลบ */
    N_pos=0; /* N_pos เก็บจำนวนข้อมูลที่เป็นบวก */
    N_neg= ; /* N_neg เก็บจำนวนข้อมูลที่เป็นลบ */
    do
    {
        scanf("%d",&data);
        if(data!= )
        {
            if( )
            {
                sum_pos= ;
                N_pos= ;
            }
            else
            {
                sum_neg= ;
                N_neg= ;
            }
        }
    }
    while(data!= );

    avg_pos=(double) sum_pos/N_pos;
    avg_neg= ;
    printf("%.2f\n",avg_pos);
    printf( );

    return 0;
}
```

8. Problem name: Reverse_Number

จงเขียนโปรแกรมที่ทำการสลับลำดับของตัวเลขแบบจำนวนเต็มบวก (Positive integer) ถูกป้อนเข้ามาทางคีย์บอร์ด แล้วแสดงตัวเลขนั้นออกสู่หน้าจอ ตัวอย่างเช่น ถ้าตัวเลขที่ถูกป้อนเข้ามาคือ 8735 ผลลัพธ์ที่แสดงออกสู่หน้าจอคือ 5378

ตัวอย่างการทำงานของโปรแกรม

ข้อมูลนำเข้า (Input)	ข้อมูลส่งออก (Output)
12345678	87654321
8375	5738
12300	00321
9	9

Hints:

(1)(a) $8375 \% 10 = 5$ $8375/10=837$

$837 \% 10 = 7$ $837/10=83$

$83 \% 10 = 3$ $83/10=8$

$8 \% 10 = 8$ $8/10=0$

(b) $12300 \% 10 = 0$ $12300/10=1230$

$1230 \% 10 = 0$ $1230/10=123$

$123 \% 10 = 3$ $123/10=12$

$12 \% 10 = 2$ $12/10=1$

$1 \% 10 = 1$ $1/10=0$

(2) Use a do-while statement and continuously strip off and display the unit digits of the number.

```
#include <stdio.h>
int main()
{
    int num, digit;

    scanf("%d", &num);
    do
    {
        digit =          ; /* digit = หลักหน่วยของข้อมูลปัจจุบันใน num */
        num =          ; /* update ค่า num ลดค่า num ไป 1 หลัก เช่น 943 -> 94 */
        printf("%d", digit);
    }
    while (          );
    printf("\n");
    return 0;
}
```