

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

Programming Homework 10

กำหนดส่งภายในวันจันทร์ที่ 12 ก.พ. 2555 เวลา 23.59 น.

คำเตือน

นิสิตสามารถจับกลุ่มปรึกษาหารือกันได้ แต่นิสิตแต่ละคนในกลุ่มจะต้องลงมือเขียนโปรแกรมด้วยตนเองและส่งโปรแกรมของตนเองโดยไม่ขึ้นอยู่กับสมาชิกในกลุ่ม และอย่านำโปรแกรมของตนเองให้กับผู้อื่นดู
ถ้าตรวจพบว่านิสิตนำงานของคนอื่นมาส่งหรือนิสิตก๊อปปี้ 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: Extract_URL_stdin

จงทำโปรแกรมข้างล่างให้สมบูรณ์เพื่อให้โปรแกรมทำการประมวลผลข้อความนำเข้าที่อ่านตัวอักษรเข้ามาทีละตัวโดยใช้ฟังก์ชัน `fgetc(stdin)` และค้นหาสตริงที่อยู่ในรูปแบบ `"http://*"` (รวมเครื่องหมาย “ ด้วย) โดยที่ * หมายถึงตัวอักษรที่ตัวก็ได้ ดังตัวอย่างเช่น

“`http://www.swu.ac.th`” โดยที่ * หมายถึง `www.swu.ac.th`

“`http://www.google.com`” โดยที่ * หมายถึง `www.google.com`

ตัวอย่างการทำงานของโปรแกรม

ข้อมูลนำเข้า (Input)	ข้อมูลส่งออก (Output)
<code>I am a "SWU"</code>	<code>"http://www.swu.ac.th"</code>
<code>Google "Mail".</code>	<code>"http://www.gmail.com"</code>
<code>We are http://123.com.</code>	<code>"http://www.123.com."</code>
<code>CAFE GRADER @ SWU "http" </code>	<code>"http://cslab.swu.ac.th/grader/"</code>
<code>URL ="http://123.145.167.89:port=8080" Here we "go"</code>	<code>"http://123.145.167.89:port=8080"</code>

Hints

(1) `stdin` เป็น system file stream ซึ่งการอ่านข้อมูลจากไฟล์นี้ เช่น `fgetc(stdin)` จะเทียบเท่ากับการอ่านข้อมูลนำเข้า 1 ตัวอักษรจากคีย์บอร์ด

(2) โปรแกรมข้างล่างนี้ทำการค้นหาตัวอักษรที่มีรูปแบบ `"http://*"` ซึ่งทำงานตาม state diagram ซึ่งมีทั้งหมด 8 state คล้ายกับโจทย์ข้อ What_in_Parentheses

โปรแกรมตัวอย่าง (บางส่วน)

```
#include <stdio.h>
```

```
#define MAX_STR_LEN 500
```

```

int main()
{
    char URL[MAX_STR_LEN+1], ch;
    int N, state;

    state=0;
    while(1)
    {
        ch=fgetc(stdin);
        if(ch=='\n')
            break;

        if(state==0)
        {
            if(ch=='')
                state=1;
        }
        else if(state==1)
        {
            if(ch=='h');
            state=...;
        }
        else if(state==2)
        {
            if(ch==...);
            state=3;
        }

        ...

        ...

        else if(state==8)
        {
            if(ch!='')
            {
                URL[N]=...;
                N=...
            }
            else
            {
                URL[N]=...;
                printf("\"http://%s\"\n", URL);
                state=0;
            }
        }
    }

    return 0;
}

```

2. Problem name: Find_students

กำหนดให้ข้อมูลนำเข้าเป็นข้อมูลของนักเรียนที่ลงทะเบียนเรียนในวิชาหนึ่งๆ โดยที่บรรทัดแรกคือจำนวนนักเรียนทั้งหมดสมมติว่าเท่ากับ N ($N \leq 50$) และในแต่ละบรรทัดของบรรทัดที่ 2 ถึงบรรทัดที่ $N+1$ คือข้อมูลของนักเรียนแต่ละคนซึ่งประกอบไปด้วย รหัส ชื่อ เพศ คณะ ชั้นปี และ คะแนนสอบ โดยที่รหัส, ชั้นปี และคะแนนสอบ คือตัวเลขจำนวนเต็ม, ชื่อและคณะคือสตริงที่มีความยาวไม่เกิน 30 ตัวอักษร, เพศคือตัวอักษร F (เพศหญิง) หรือ M (เพศชาย)

จงเขียนโปรแกรมเพื่อค้นหาและแสดงข้อมูลของนักเรียนผู้ชาย ที่อยู่ชั้นปีที่ 1 ที่มีคะแนนสอบมากกว่าหรือเท่ากับคะแนนเฉลี่ยของคะแนนสอบของนักเรียนทุกคน โดยให้แสดงผลลัพธ์เป็นคะแนนเฉลี่ย (จุดทศนิยม 2 ตำแหน่ง) ในบรรทัดแรก และแสดงข้อมูลนักเรียนแต่ละคนที่ค้นเจอที่ละบรรทัด ตามลำดับที่ข้อมูลที่ถูกป้อนเข้ามา โดยสมมติว่าในชุดของข้อมูลทดสอบจะมีคำตอบอย่างน้อย 1 คนเสมอ

ตัวอย่างการทำงานของโปรแกรม

ข้อมูลนำเข้า (Input)	ข้อมูลส่งออก (Output)
10 10101 John M SCI 2 60 10009 Sarah F ECENG 2 50 10001 Abby M MED 1 100 10084 Onny M COMSCI 2 90 11101 Jane M ECON 1 50 12009 Soe F DENT 2 80 12001 Tab M MUSIC 1 80 13084 Anny F BUSINESS 3 50 13184 Peter M BUSINESS 1 40 13284 Bob M HUMANSKI 2 40	64.00 10001 Abby M MED 1 100 12001 Tab M MUSIC 1 80

3. Problem name: Sort_GPA_student

จงเขียนโปรแกรมเพื่อทำการเรียงและแสดงข้อมูลของนักเรียนตามคะแนนเกรดเฉลี่ยจากมากไปน้อย โดยกำหนดให้ข้อมูลนำเข้าบรรทัดแรกคือจำนวนนักเรียนทั้งหมดสมมติว่าเท่ากับ N ($N < 50$) และในแต่ละบรรทัดของบรรทัดที่ 2 ถึงบรรทัดที่ N+1 คือข้อมูลของนักเรียนแต่ละคนซึ่งประกอบไปด้วย รหัส ชื่อ นามสกุล เพศ อายุ และเกรดเฉลี่ย นอกจากนี้กำหนดให้เขียนโปรแกรมโดยใช้ตัวแปรแบบ **structure** ที่เหมาะสม

ตัวอย่างการทำงานของโปรแกรม

ข้อมูลนำเข้า (Input)	8 530010101 John Jane M 25 3.13 530010009 Sarah Lee F 19 2.98 530010001 Abby Mater M 17 3.10 530010084 Onny Lynch F 21 3.95 530011101 Jane John M 25 2.13 530012009 So Lee-Tung F 20 3.98 530012001 Tab Mccoy M 18 2.10 530013084 Anny Lunch F 22 2.95
ข้อมูลส่งออก (Output)	530012009 So Lee-Tung F 20 3.98 530010084 Onny Lynch F 21 3.95 530010101 John Jane M 25 3.13 530010001 Abby Mater M 17 3.10 530010009 Sarah Lee F 19 2.98 530013084 Anny Lunch F 22 2.95 530011101 Jane John M 25 2.13 530012001 Tab Mccoy M 18 2.10

4. Problem name: Closest_pairs_of_3D_points

กำหนดให้ข้อมูลนำเข้าซึ่งเป็นข้อมูลของเซตของจุดในปริภูมิ 3 มิติ (3D points) มีรูปแบบดังนี้

```
N
<id> <x> <y> <z>
...
<id> <x> <y> <z>
```

โดยที่

N หมายถึงจำนวนจุดใน 3D Space ที่เก็บอยู่ในไฟล์

id หมายถึงรหัสหมายเลขของจุด

x, y, z หมายถึง ตำแหน่งโคออร์ดิเนต x, y, z ของจุดเมื่อเทียบกับจุด origin (0, 0, 0)

จงเขียนโปรแกรมเพื่ออ่านข้อมูลนำเข้าให้นำมาเก็บใน Dynamically sized array of structures ที่ถูกสร้างขึ้น ณ ขณะ Run-time โดยมีจำนวน element เท่ากับ N และกำหนดให้ structure ที่แทนข้อมูลแต่ละจุดถูกนิยามดังนี้

```
typedef struct{
    int id;
    double x;
    double y;
    double z;
} Point3D;
```

หลังจากนั้นให้ทำการคำนวณหาว่าแต่ละจุด ใน Point set มีระยะห่างใกล้กับจุดใดใน Point set (ยกเว้นตัวมันเองซึ่งระยะห่างจะเป็น 0) และให้แสดงผลลัพธ์ในรูปแบบดังนี้

<Point1_ID>	<Closest_Point_ID>
<Point2_ID>	<Closest_Point_ID>
...	...
<PointN_ID>	<Closest_Point_ID>

ตัวอย่างการทำงานของโปรแกรม

ข้อมูลนำเข้า (Input)	ข้อมูลส่งออก (Output)
15 1 2.8864 -8.8076 -1.5423 2 -2.4278 3.6394 -8.1154 3 6.2316 -9.1514 1.9705 4 0.6565 -8.5711 -0.5815 5 -2.9855 0.4330 3.9190 6 8.7800 -8.0654 3.9978 7 7.5189 6.3630 2.7706 8 1.0031 6.3509 -9.3279 9 2.4495 4.4488 -8.6239 10 1.7409 -7.0027 -3.6080 11 -5.8452 3.1921 0.6173 12 -3.9751 0.3719 3.0889 13 -0.5815 9.4595 -1.8476 14 -5.3902 2.9798 6.3996 15 6.8862 6.0066 4.3672	1 4 2 8 3 6 4 1 5 12 6 3 7 15 8 9 9 8 10 1 11 12 12 5 13 8 14 5 15 7

หมายเหตุ

- การทดสอบโปรแกรมให้ใช้ Menu Copy | Paste บน Command Prompt ในการป้อนข้อมูลนำเข้า โดย Test case ของข้อนี้สามารถดาวน์โหลดที่ Course website
- ตัวอย่างข้างบน ผลลัพธ์ที่ได้คือ จุดที่ 1 จะใกล้กับจุดที่ 4 ที่สุด, จุดที่ 2 จะใกล้กับจุดที่ 8 ที่สุด และ จุดที่ 3 จะใกล้กับจุดที่ 6 ที่สุด

5. Problem name: Rectangle_Area_Ratio

กำหนดให้ข้อมูลนำเข้าคือเซตของสี่เหลี่ยมผืนผ้า (Rectangle) ซึ่งมีรูปแบบต่อไปนี้

N						<- Numbers of rectangles
ID	Sx	Sy	Ex	Ey		<- Rectangle 1
.....						
ID	Sx	Sy	Ex	Ey		<- Rectangle N

โดยที่ N หมายถึง จำนวนข้อมูลของสี่เหลี่ยมผืนผ้า (Rectangle) ที่เก็บอยู่ในไฟล์

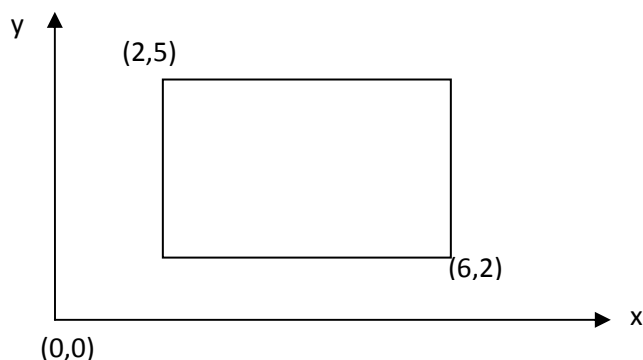
ID หมายถึง รหัสหมายเลขของ Rectangle หนึ่งๆ

Sx, Sy หมายถึง XY-Coordinate ของจุดซ้ายบน (upper left) ของ Rectangle

Ex, Ey หมายถึง XY-Coordinate ของจุดขวาล่าง (lower right) ของ Rectangle

และกำหนดให้ Sx, Sy, Ex, Ey เป็นค่าจำนวนเต็มที่ไม่มากกว่าหรือเท่ากับ 0

ตัวอย่าง เช่น Rectangle ข้างล่างนี้มีค่า (Sx, Sy) เท่ากับ (2, 5) และ (Ex, Ey) เท่ากับ (6, 2) ซึ่งพื้นที่จะมีค่าเท่ากับ $(6-2) \times (5-2) = 12$



จงเขียนโปรแกรมเพื่ออ่านข้อมูลนำเข้าให้นำมาเก็บใน Dynamically sized array of structures ที่ถูกสร้างขึ้น ณ ขณะ Run-time โดยมีจำนวน element เท่ากับ N และกำหนดให้ structure ที่แทนข้อมูลแต่ละ Rectangle ถูกนิยามดังนี้

```
typedef struct{
    int id;
    int sx,sy,ex,ey;
    int area;    // This field is computed from sx,sy,ex,ey
} Rectangle;
```

หลังจากนั้นให้คำนวณอัตราส่วนระหว่างพื้นที่ของแต่ละ Rectangle หารด้วยผลรวมของพื้นที่ Rectangle ทั้งหมด

โปรแกรมจะแสดงผลเป็นข้อมูลของแต่ละ Rectangle ที่รับเข้ามา ค่าพื้นที่ของ Rectangle และ อัตราส่วนพื้นที่ที่คำนวณได้

ตัวอย่างการทำงานของโปรแกรม

ข้อมูลนำเข้า (Input)	ข้อมูลส่งออก (Output)
10 1 3 7 9 3 2 1 10 4 6 3 1 4 6 1 4 6 12 10 6 5 8 4 11 1 6 10 7 14 3 7 11 10 13 2 8 7 10 9 7 9 1 13 5 8 10 3 5 14 4	1 3 7 9 3 24 0.16 2 1 10 4 6 12 0.08 3 1 4 6 1 15 0.10 4 6 12 10 6 24 0.16 5 8 4 11 1 9 0.06 6 10 7 14 3 16 0.10 7 11 10 13 2 16 0.10 8 7 10 9 7 6 0.04 9 1 13 5 8 20 0.13 10 3 5 14 4 11 0.07

หมายเหตุ

- การทดสอบโปรแกรมให้ใช้ Menu Copy | Paste บน Command Prompt ในการป้อนข้อมูลนำเข้า โดย Test case ของข้อนี้สามารถดาวน์โหลดที่ Course website
- จากตัวอย่างข้างบน ค่าพื้นที่ของ Rectangle #1 ที่คำนวณได้คือ 24 และ คิดเป็นอัตราส่วน 0.16 ของพื้นที่ทั้งหมด

6. Problem name: Count_Article

จงเขียนโปรแกรมเพื่อนับจำนวน article 'a', 'an', 'the' ที่ปรากฏอยู่ในประโยคในภาษาอังกฤษหลายๆ ประโยคซึ่งเป็นข้อมูลนำเข้า โดยที่ตัวอักษรเล็กหรือใหญ่จะถูกพิจารณาเป็นตัวเดียวกัน เช่น A จะเหมือนกับ a, An จะเหมือนกับ an, aN, AN และ the และเหมือนกับ The และ tHe เป็นต้น กำหนดให้สตริงนำเข้าจะมีความยาวไม่เกิน 1000 ตัวอักษร และให้แสดงผลลัพธ์เป็นจำนวน A, จำนวน An และ จำนวน The บรรทัดละค่าเรียงต่อกันมา

ตัวอย่างการทำงานของโปรแกรม

ข้อมูลนำเข้า (Input)	ข้อมูลส่งออก (Output)
The computing world has undergone a revolution since the publication of The C Programming Language in 1978 Big computers are much bigger and personal computers have capabilities that rival mainframes of a decade ago During this time C has changed too although only modestly and it has spread far beyond its origins as the language of the UNIX operating system	2 0 5
I work in a factory Can I have an apple She is an English teacher The car over there is fast The house is quite old and has four bedrooms An ant is walking along The road in side A ban out of the an	2 4 4

Hint: ใช้โค้ดที่เขียนในข้อ Extract_Word ดึงคำแต่ละคำออกมาจากสตริงอินพุต ในแต่ละคำ เช่น "tHe" ให้แปลงเป็นตัวอักษรตัวเล็กทั้งหมด ("the") หลังจากนั้นให้ใช้ฟังก์ชัน string_compare ที่เขียนใน homework ก่อนหน้านี้ทำการเปรียบเทียบว่าคำแต่ละคำนั้นคือ "a", "an", "the"