

ปล. อาจมีบ้างข้อเฉลยพลาดได้
เพราะใช้เวลาอันสั้นในการเฉลย
หากมีข้อผิดพลาด แจ้งฟีดแบ็ก



สอบปลายภาค: ภาคการศึกษาที่ 1

ปีการศึกษา: 2559

วันสอบ: 11 ธันวาคม 2559

เวลาสอบ: 13.30 – 16.30 น.

ห้องสอบ: A400 (01,03) A401(04,05,07) R201(08,09) S817(06, 241-101:01, 242-101:02)

หัวหูน (02, 242-101: 01)

ผู้สอน: อ.สุกัญญา อ.อารีย์ อ.เสกสรรค์ อ.วรพต อ.วศิน อ.นิคม ภาควิชาวิศวกรรมคอมพิวเตอร์

รหัสและชื่อวิชา: 240-101, 241-101, 242-101 Introduction to Computer Programming

แนะนำการเขียนโปรแกรมคอมพิวเตอร์

ทูลรติในการสอบมีโทษขั้นต่ำคือ ปรับตกในรายวิชาที่ทูลรติและพักการเรียน 1 ภาคการศึกษา

คำสั่ง: อ่านรายละเอียดของข้อสอบ และคำแนะนำให้เข้าใจก่อนเริ่มทำข้อสอบ

อนุญาต: เครื่องเขียนต่าง ๆ เช่น ปากกา หรือดินสอ เข้าห้องสอบ

ไม่อนุญาต: หนังสือ หรือเครื่องคิดเลขเข้าห้องสอบ และเอกสารใด ๆ เข้าและออกห้องสอบ

เวลา: 3 ชั่วโมง (180 นาที)

คำแนะนำ

- ☐ ข้อสอบมี 13 หน้า (รวมหน้าปก) แบ่งออกเป็น 3 ตอน คะแนนรวม 90 คะแนน (คิดเป็นคะแนนเก็บ 45%)
- ☐ เขียนคำตอบในข้อสอบ คำตอบส่วนใดอ่านไม่ออก จะถือว่าคำตอบนั้นผิด
- ☐ อ่านคำสั่งในแต่ละข้อให้เข้าใจก่อนลงมือทำ
- ☐ เวลาที่ใช้เวลาทำตอนให้เหมาะสม ตามคำแนะนำ
- ☐ หากข้อใดเขียนคำตอบไม่พอ ให้เขียนเพิ่มเติมด้านหลังของหน้านั้นเท่านั้น

	1	2	3	รวม
ตอน	(40)	(40)	(10)	(90)
	20%	20%	5%	45%
คะแนน				

นักศึกษารับทราบ ลงชื่อ 6 / 12 / 2560 By.ฟิสิกส์

ตอนที่ 1 ฟังก์ชัน Function (40 คะแนน / 75 นาที)

ข้อ 1. จงเลือกต้นแบบของฟังก์ชัน (function prototype) ให้เหมาะสมกับการเรียกใช้ (10 คะแนน)

Function prototype ต้นแบบของฟังก์ชัน

ตัวเลือก	ต้นแบบของฟังก์ชัน
A	int fun(int x)
B	int fun(char x, float y)
C	float fun(void)
D	float fun(float x, int y)
E	void fun(int x[], float y)
F	char fun(char x[], int x)

ส่วนของโปรแกรมและการเรียกใช้ฟังก์ชัน	คำตอบ (ใส่เฉพาะตัวเลือก A-F) (เลือกฟังก์ชัน function prototype ที่เหมาะสมที่สุด)
int a=3, b=5, c[5]={0}; char p[]="Hello"; float m=1.25;	
printf("%d %d \n", a+1, fun(a+1));	A
m = fun(m, 5);	D
a = 2 + fun(p[1], 2.5) * 3;	B
printf("%.3f \n", fun());	C
fun(c, 3.14159);	E

ข้อ 2. กำหนดนิยามของฟังก์ชัน F และ G ให้หาผลลัพธ์จากส่วนของโปรแกรม (10 คะแนน)

#include <string.h> int F(char x[], char y) { int n, a=0; for(n=0;n<strlen(x);n++) if(x[n]==y) a++; return a; }	#include <string.h> int G(char x[], char y[]) { strcat(x, "+"); strcat(x, y); a=strlen(x); return a; }
ส่วนของโปรแกรมและการเรียกใช้ฟังก์ชัน	ผลลัพธ์จากส่วนของโปรแกรม
int a=3, b=4; char x[100]="Hello"; char p[20]="New World";	
printf(" %d \n", strlen(x));	5
printf(" %d \n", strcmp(x, "Hello"));	0

<code>printf(" %d \n", F(p, 'e'));</code>	1
<code>printf(" %d \n", F("Goodbye", x[4]));</code>	2
<code>printf(" %d \n", G(x, p));</code>	15

ข้อ 3. จงเขียนโปรแกรมคำนวณค่าบริการจอดรถยนต์ (10 คะแนน)



รูปที่ 1. อัตราค่าบริการจอดรถยนต์ ณ อาคารใกล้โรงพยาบาลแห่งหนึ่ง
ในมหาวิทยาลัยแห่งหนึ่งในภาคใต้

โปรแกรมจะวนรับข้อมูล จำนวนชั่วโมง และ นาที ของการจอดรถคันหนึ่งๆ เพื่อคำนวณค่าบริการแล้วแสดงผล
เสร็จแล้วจึงรอรับข้อมูลถัดไป จนกระทั่งเมื่อผู้ใช้ใส่ค่าเป็น 0 ชั่วโมง และ 0 นาที โปรแกรมจะจบการทำงาน

ตัวอย่างการรันโปรแกรม (ตัวเลข **เอียงหนา** คือค่าที่รับจากผู้ใช้)

```
Enter parking duration (hours and minutes): 0 30
Parking cost = 50 baht
Enter parking duration (hours and minutes): 4 14
Parking cost = 50 baht
Enter parking duration (hours and minutes): 4 15
Parking cost = 60 baht
Enter parking duration (hours and minutes): 7 0
Parking cost = 80 baht
Enter parking duration (hours and minutes): 0 0
End of program.
```

กำหนดให้ ในโปรแกรมมีฟังก์ชัน calculate ที่คำนวณค่าบริการ ซึ่งฟังก์ชันรับค่า จำนวนชั่วโมงและนาที ผ่านพารามิเตอร์สองตัว แล้วคืนค่ากลับเป็นค่าบริการที่คำนวณได้

จงเติมโปรแกรมนี้ให้สมบูรณ์

```
#include <stdio.h>
int calculate(int hr, int mn);
int main( )
{
    int hr, mn, cost;
    do
    { printf("parking duration (hours and minutes): ");
      //รับค่าจำนวนชั่วโมงและนาที เก็บในตัวแปร hr และ mn

      scanf("%d %d",&hr,&mn);

      // ถ้า hr และ mn มีค่าเป็น 0 ให้ออกจากลูป do-while ทันที ด้วยคำสั่ง break
      if(      hr==0 && mn==0      )

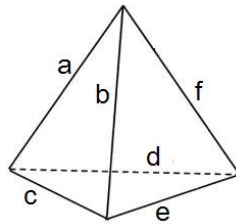
          break;

      // คำนวณค่าบริการ โดยเรียกใช้ฟังก์ชัน calculate
      cost =      calculate(hr,mn);
      printf("Parking cost = %d\n", cost);

    } while (      hr!=0 || mn!=0      ); // เงื่อนไขของลูป do-while

    printf("End of program.");      or      !(hr!=0 || mn!=0)
    return 0;
}
// เขียนนิยามของฟังก์ชัน calculate ให้สมบูรณ์
int calculate(int hr, int mn)
{
    int sum = 0;
    if(hr==0&&mn<15){ return 0;}
    else{
        if(mn>=15){
            hr++;
        }
        if(hr<=4){
            sum = 50;
        }
        else{
            sum = 50 + (hr-4)*10;
        }
    }
    return sum;
}
```

ข้อ 4. จงเขียนโปรแกรมคำนวณพื้นที่ผิวของรูป Tetrahedron (10 คะแนน)



รูปที่ 2. Tetrahedron

Tetrahedron คือรูปทรงเรขาคณิตสามมิติที่ประกอบขึ้นด้วยรูปสามเหลี่ยม 4 รูป พื้นที่ผิวของรูป Tetrahedron จึงหาได้จาก ผลรวมของพื้นที่ของรูปสามเหลี่ยมทั้งสี่รูป ตัวอย่าง เช่น รูป Tetrahedron ที่มีด้าน a, b, c, d, e และ f จะมีพื้นที่ผิวทั้งหมด คือ ผลรวมของพื้นที่รูปสามเหลี่ยมสี่รูป ได้แก่ Δabc , Δadf , Δbef และสามเหลี่ยม Δcde จงเขียนโปรแกรมรับความยาวด้านของรูป Tetrahedron ทั้ง 6 ด้าน (เรียกใช้ฟังก์ชัน inputSides) แล้วคำนวณหาพื้นที่ผิว (Surface Area) กำหนดให้โปรแกรมสร้างฟังก์ชัน float Triangle(float a, float b, float c) เพื่อใช้หาพื้นที่รูปสามเหลี่ยมหนึ่งรูปจากความยาวด้าน 3 ด้าน (a, b, c) โดยพื้นที่รูปสามเหลี่ยม (A) หาได้จากสูตร

$$A = \sqrt{S(S-a)(S-b)(S-c)} \quad \text{เมื่อ} \quad S = \frac{a+b+c}{2}$$

จงเติมโปรแกรมนี้ให้สมบูรณ์

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

```
void inputSides(float s[6]){
    int j;
    printf("Input the 6 sides of tetrahedron: ");
    for(j=0; j<6; j++) scanf("%f", &s[j]);
}
```

//นิยามของฟังก์ชัน

```
float Triangle(float a, float b, float c)
{
```

```
.....
float S,A;
.....
```

```
S = (a+b+c)/2;
.....
```

```
.....
A = sqrt(S*(S-a)*(S-b)*(S-c));
return A
.....
.....
```

```
}
```


ตอนที่ 2 สตรักเจอร์ Structure (40 คะแนน / 75 นาที)

ข้อ 1. จากสตรักเจอร์เกี่ยวกับรายละเอียดหนังสือ (ชื่อหนังสือ, ชื่อผู้แต่ง, จำนวนสำเนาของหนังสือรายการนั้นๆ ว่ามีจำนวนกี่เล่มในห้องสมุด) และตำแหน่งที่เก็บหนังสือ (หมวดหนังสือ, ชั้นของอาคาร, และ หมายเลขชั้นวางหนังสือ) ของหนังสือ 3 รายการ ในห้องสมุดที่กำหนด จงเขียนบางส่วนของโปรแกรมดังนี้ (10 คะแนน : 10 นาที)

```
struct book_location { char section[10]; int floor, shelf_number; }
struct book_detail { char book_title[30], author[30]; int num; struct book_location location; } book1, book2, book3;
```

1.1 รับค่าชื่อของหนังสือรายการที่ 1 จากผู้ใช้

```
scanf("%s", book1.book_title);
```

1.2 กำหนดชื่อหนังสือรายการที่ 2 ให้มีชื่อว่า Sherlock Holme

```
strcpy(book2.book_title, "Sherlock Holme");
```

1.3 กำหนดให้หนังสือรายการที่ 3 มีจำนวนสำเนาทั้งหมด 10 เล่ม ในห้องสมุด

```
book3.num = 10;
```

1.4 แสดงออกทางหน้าจอว่า จำนวนสำเนาของหนังสือทั้ง 3 รายการ มีทั้งหมดกี่เล่ม

```
printf("%d", book1.num+book2.num+book3.num);
```

1.5 รับค่าหมายเลขชั้นวางหนังสือ ของหนังสือรายการที่ 1

```
scanf("%d", &book1.lacation.shelf_number);
```

ข้อ 2. จากบางส่วนของโปรแกรม ที่มีการประกาศตัวแปรชนิดใหม่เพื่อเก็บข้อมูลพนักงานในแผนก ซึ่งได้แก่ชื่อ ตำแหน่ง อายุ และเงินเดือน ให้นักศึกษาเติมโปรแกรมให้สมบูรณ์ (10 คะแนน : 10 นาที)

```
int main ()
{ int i, total=0;
  float average;
  typedef struct
  { char name[30], position[30];
    int age;
    float salary;
  } person;
```

***จงประกาศตัวแปรอาร์เรย์ person ชื่อ staff สำหรับเก็บข้อมูลพนักงาน 3 คนในแผนก พร้อมทั้งกำหนด (ไม่ใช่รับค่าจากผู้ใช้) ข้อมูลพนักงานแต่ละคนดังนี้**

พนักงานคนที่ 1 ชื่อ John Luca ตำแหน่ง Director อายุ 55 ปี เงินเดือน 80,000 บาท

พนักงานคนที่ 2 ชื่อ Top Smith ตำแหน่ง Programmer อายุ 35 ปี เงินเดือน 55,000 บาท

พนักงานคนที่ 3 ชื่อ Susan Tippy ตำแหน่ง Programmer อายุ 25 ปี เงินเดือน 30,000 บาท

```
..... person staff[3] = {{"John Luca","Director",55,80000},
..... {"Top Smith","Programmer",35,55000},
..... {"Susan Tippy","Programmer",25,30000}
..... };
.....
```

//จงเขียนโปรแกรม **วน loop** หาอายุเฉลี่ยของพนักงานทั้ง 3 คน

```
..... int i;
..... for(i=0;i<3;i++)
.....     total += staff[i].age;
.....     average = total/3;
.....
```

```
printf("Average age is %.2f", average);
```

//Top Smith ได้รับการปรับตำแหน่งเป็น Senior Programmer และเงินเดือนเพิ่มขึ้นเป็น 60,000 บาท

```
..... strcpy(staff[1].position,"Senior Programmer");
..... staff[1].salary = 60000;
.....
```

ข้อ 3. พนักงานร้านล้างรถแห่งหนึ่งนำไปเสร็จค่าล้างรถของวันนี้มากกรอกลงในโปรแกรมเพื่อบันทึกรายละเอียดของรถที่เข้ามาใช้บริการ โดยใช้งานตัวแปรชนิดใหม่ตามที่กำหนด ให้ นศ. เขียนโปรแกรมเพื่อรับรายละเอียดต่างๆ ของรถที่เข้ามาใช้บริการ (กำหนดให้รับลูกค้าได้สูงสุด 50 คัน) จากนั้นให้โปรแกรมค้นหาและแสดง “ป้ายทะเบียนรถยนต์คันแรก” ที่เข้ามาใช้บริการ (10 คะแนน : 20 นาที)

```
typedef struct
{   char license_plate[10];
    int hour, min;
} details;
```

ตัวอย่างผลการรันโปรแกรม (**ตัวอักษรหนา** คือตัวอย่างค่าที่ผู้ใช้ป้อนลงในโปรแกรม)

```
Enter License Plate : AA2566
Enter entry time (hour:min) : 9:30
Enter License Plate : XY1234
Enter entry time (hour:min) : 8:45
The first car is XY1234
```



```
#include<stdio.h>
typedef struct
{
    char license_plate[10];
    int hour, min;
} details;

void input(details c[],int n){
    int i;
    for(i=0;i<n;i++){
        printf("Enter License Plate : ");
        scanf("%s",c[i].license_plate);
        printf("Enter entry time (hour:min) :");
        scanf("%d:%d",&c[i].hour,&c[i].min);
    }
}

details Calfirst(details c[],int n){
    int i;
    details min = c[0];
    for(i=0;i<n;i++){
        if(c[i].hour < min.hour){
            min = c[i];
        }
        else if(c[i].hour == min.hour){
            if(c[i].min < min.min){
                min = c[i];
            }
        }
    }
    return min;
}

int main(){
    details c[50],min;
    int n=3;
    input(c,n);
    min = Calfirst(c,n);
    printf("The first car is %s",min.license_plate);
    return 0;
}
```

**หมายเหตุจำเป็นต้องทำเป็นฟังก์ชันนะ
สามารถทำใน main หหมดดีได้

ค่าของ n เป็นสมมติเอา

ข้อ 4. จงเขียนโปรแกรมเพื่อให้ครูชั้นมัธยมศึกษาปีที่ 4 สามารถกรอก ชื่อ รหัสนักศึกษา และคะแนนสอบวิชาต่างๆ ของ นร. ทั้ง 30 คนลงไปได้ในฟังก์ชัน main จากนั้นใช้งานฟังก์ชัน show_data ในการแสดงรายละเอียดของ นักศึกษาแต่ละคนออกทางหน้าจอ โดยมี function prototype ดังนี้

```
void show_data(struct student classM4[30]);
```

ในการเขียนโปรแกรม กำหนดให้มีการใช้งานสตรัคเจอร์ต่างๆ ดังนี้

<pre>struct score { int maths, eng, physics; };</pre>	<pre>struct student { char name[30]; char id[10]; struct score scr; };</pre>
---	--

***นศ. สามารถประกาศตัวแปรเพิ่มเติมได้ (10 คะแนน : 20 นาที)

ผลการรันโปรแกรม (ตัวอักษรหนา คือตัวอย่างค่าที่ผู้ใช้ป้อนลงในโปรแกรม)

```
Enter name of student 1 : Thana
Enter id of student 1 : 59COE0001
Enter all scores of student 1 : 70 75 88
:
Enter name of student 30 : Jinda
Enter id of student 30 : 59COE0030
Enter all scores of student 30 : 66 80 70
Show All Students:
Name      ID      Maths      Eng      Physics
Thana     59COE0001  70        75      88
:
Jinda     59COE0030  66        80      70
```

```
.....
#include<stdio.h>
.....
struct score
.....
{ int maths, eng, physics;
.....
};
.....
struct student
.....
{ char name[30];
.....
  char id[10];
.....
  struct score scr;
.....
};
.....
void show_data(struct student classM4[30]);
.....
.....
.....
.....
.....
.....
.....
.....
.....
```

```
int main(){
    int i;
    struct student st[30];
    for(i=0;i<30;i++){
        printf("Enter name of student %d :",i+1);
        scanf("%s",st[i].name);
        printf("Enter id of student %d :",i+1);
        scanf("%s",st[i].id);
        printf("Enter all scores of student %d :",i+1);
        scanf("%d %d %d",&st[i].scr.maths,
                &st[i].scr.eng,&st[i].scr.physics);
    }
    show_data(st);
}

void show_data(struct student classM4[30]){

    int i;
    printf("Show All Students : \n");
    printf("Name ID Maths Eng Physics\n");
    for(i=0;i<30;i++){
        printf("%s %s %d %d %d\n",
                classM4[i].name,classM4[i].id,classM4[i].scr.maths,
                classM4[i].scr.eng,classM4[i].scr.physics);
    }
}
```

/****** จบตอนที่ 2 *****/

ตอนที่ 3 โปรแกรมประยุกต์ (10 คะแนน / 30 นาที)

มีการประกาศชนิดข้อมูล Date ไว้สำหรับเก็บวันที่ และ Friend เป็นโครงสร้างข้อมูลเก็บชื่อและวันเกิดของเพื่อน ดังนี้

```
typedef struct {
    int dd; //day วันที่
    int mm; //month เดือน
    int yy; //year ปี
} Date;
typedef struct {
    char name[50]; //ชื่อ
    char telephone[15]; // เบอร์โทรศัพท์
    Date birthdate; //วันเกิด
} Friend;
```

จงเขียนนิยามของฟังก์ชัน ต่อไปนี้ (เขียนเฉพาะนิยามฟังก์ชัน ไม่ต้องเขียนโปรแกรมทั้งหมด)

ข้อ 1. นิยามฟังก์ชัน birthMonth เป็นฟังก์ชันพิมพ์รายชื่อของเพื่อนที่เกิดในเดือนใดเดือนหนึ่ง (5 คะแนน)

ฟังก์ชันมี prototype ดังนี้

```
void birthMonth(Friend Fds[ ] , int n, int m)
```

พารามิเตอร์ Fds คือ อาร์เรย์ชนิด Friend เก็บข้อมูลของเพื่อนๆ และ n คือจำนวนข้อมูลในอาร์เรย์ ส่วน m คือ เดือนที่ใช้ค้นหาวันเกิด ฟังก์ชันนี้จะพิมพ์รายชื่อเพื่อนทุกคนที่มีวันเกิดในเดือนที่ตรงกับค่า m ฟังก์ชันไม่มีการคืนค่า

```
.....
void birthMonth(Friend Fds[ ] , int n, int m){
.....
    int i;
.....
    for(i=0;i<n;i++){
.....
        if(Fds[i].birthdate.mm == m){
.....
            printf("%s \n",Fds[i].name);
.....
        }
.....
    }
.....
}
```

.....

.....

.....

.....

.....

ข้อ 2. นิยามฟังก์ชัน `oldestFriend` เป็นฟังก์ชันค้นหาเพื่อนที่อายุมากที่สุด (5 คะแนน)

ฟังก์ชันมี prototype ดังนี้ `Friend oldestFriend(Friend Fds[], int n)`

พารามิเตอร์ `Fds` คือ อาร์เรย์ชนิด `Friend` เก็บข้อมูลของเพื่อนๆ และ `n` คือจำนวนข้อมูลในอาร์เรย์ ฟังก์ชันนี้จะคืนค่าเป็นข้อมูลชนิด `Friend` ซึ่งเป็นเพื่อนคนที่อายุมากที่สุด (วันเกิดอยู่ก่อนเพื่อนทุกคน)

```

Friend oldestFriend(Friend Fds[ ], int n){
    int i;
    Friend max = Fds[0];
    for(i=0;i<n;i++){
        if(Fds[i].birthdate.yy < max.birthdate.yy){
            max = Fds[i];
        }
        else if(Fds[i].birthdate.yy == max.birthdate.yy){
            if(Fds[i].birthdate.mm < max.birthdate.mm){
                max = Fds[i];
            }
            else if(Fds[i].birthdate.mm == max.birthdate.mm){
                if(Fds[i].birthdate.dd < max.birthdate.dd){
                    max = Fds[i];
                }
            }
        }
    }
    return max;
}

```

/****** จบตอนที่ 3 *****/