



สอบปลายภาค: ภาคการศึกษาที่ 1

ปีการศึกษา: 2558

วันสอบ: 12 ธันวาคม 2558

เวลาสอบ: 13.30 – 16.30 น.

ห้องสอบ: (01) R201, (02) A401, (03)R201, (04) S817, (05) A401, (06) S817, (07) A400,
(08) A400, (09) หัวหูน, (10) หัวหูน, (241-101) หัวหูน

ผู้สอน: อ.มัลลิกา อ.สกุณา อ.อารีย์ อ.วรวพรต อ.เสกสรรค์ อ.นิคม

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

ทฤษฎีในการสอบมีโทษขั้นต่ำคือ ปรับตกในรายวิชาที่ทุจริตและพักการเรียน 1 ภาคการศึกษา

คำสั่ง: อ่านรายละเอียดของข้อสอบ และคำแนะนำให้เข้าใจก่อนเริ่มทำข้อสอบ

อนุญาต: เครื่องเขียนต่างๆ เช่น ปากกา หรือดินสอ เข้าห้องสอบ

ไม่อนุญาต: หนังสือ หรือเครื่องคิดเลขเข้าห้องสอบ และเอกสารใดๆ เข้าและออกห้องสอบ

เวลา: 3 ชั่วโมง (180 นาที)

คำแนะนำ

- ข้อสอบมี 14 หน้า (รวมหน้าปก) แบ่งออกเป็น 3 ตอน คะแนนรวม 80 คะแนน (คิดเป็นคะแนนเก็บ 40%)
- เขียนคำตอบในข้อสอบ คำตอบส่วนใดอ่านไม่ออก จะถือว่าคำตอบนั้นผิด
- อ่านคำสั่งในแต่ละข้อให้เข้าใจก่อนลงมือทำ
- หากข้อใดเขียนคำตอบไม่พอ ให้เขียนเพิ่มเติมด้านหลังของหน้านั้นเท่านั้น

ชื่อ _____ รหัส _____ section _____

	1	2	3	รวม
ตอน	(30) 15%	(30) 15%	(20) 10%	(80) 40%
คะแนน				

นักศึกษากรรรับทราบ ลงชื่อ

ตอนที่ 1 (30 คะแนน, 15%, 70 นาที)

ฟังก์ชัน(Function)

1. จงตอบคำถามต่อไปนี้ (10 คะแนน)

1.1 จากโปรโตไทป์ของฟังก์ชัน จงเขียนส่วนของโปรแกรมเพื่อเรียกใช้งานฟังก์ชัน

โดยสามารถใช้ค่าคงที่หรือประกาศตัวแปรได้ตามความเหมาะสม (2 คะแนน)

void ant_(float x, int y); ตอบ

char man_A(char a, int b, char c); ตอบ

1.2. ถ้าหากส่งค่า m และ n ไปให้ฟังก์ชัน Ack จะมีค่าอะไรส่งกลับมาให้กับโปรแกรม (4 คะแนน)

```
#include<stdio.h>
int Ack(int m, int n)
{
    if (m == 0)
        return n + 1;
    else if (n == 0)
        return Ack(m - 1, 1);
    else
        return Ack(m - 1, Ack(m, n - 1));
}

void main()
{
    int m, n;
    printf("%d", Ack(m, n)); // โดยที่ค่า m และ n มีค่าตามข้อ 1.2.1 และ 1.2.2
}
```

1.2.1 **Ack** (0, 0) ตอบ

1.2.2 **Ack** (1, 6) ตอบ

1.3 จงเขียนนิยามฟังก์ชัน เพื่อลบตัวอักษรออกจากสตริง โดยฟังก์ชันรับค่าเข้าเป็นสตริงและตัวอักษรที่ต้องการลบออกจากสตริงเช่น กำหนดสตริงคำว่า "apple" และตัวอักษรที่ต้องการลบเป็น 'p' ให้กับฟังก์ชัน เมื่อฟังก์ชันทำงานเสร็จสิ้น สตริงที่รับค่าเข้ามาทางพารามิเตอร์จะเหลือค่าเป็น "ale" (4 คะแนน)

2. จงตอบคำถาม ตามจุดประสงค์ในแต่ละข้อดังต่อไปนี้

(12 คะแนน)

ข้อ	คำถาม	คำตอบ
2.1	<p>แก้ไขโปรแกรม โดยให้เขียนส่วนการคำนวณค่าฟังก์ชัน F เป็นฟังก์ชันในโปรแกรม โดยกำหนดให้ โปรโตไทป์ของฟังก์ชัน เป็น <code>void get_Fx(int x);</code></p> <pre data-bbox="261 533 847 1133">#include<stdio.h> int main() { int x, y; printf("F(X)=3X if X > 0\n"); printf("F(X)=10 if X <= 0\n"); printf("Enter 2 values\n"); scanf("%d%d",&x,&y); if (x > 0) printf("F(%d) is %d",x,3*x); else printf("F(%d) is 10", x); if (y > 0) printf("F(%d) is %d", y,3*y); else printf("F(%d) is 10", y); return 0; }</pre>	
2.2	<p>จงแปลงการคำนวณค่านี้ให้เป็นคำสั่งภาษา C (สามารถเรียกใช้ฟังก์ชันมาตรฐานจาก math.h ได้)</p> $v = (a^{m+n}) * \sqrt{a} + 10$	
2.3	<p>แสดงผลลัพธ์ของโปรแกรม</p> <pre data-bbox="261 1585 799 1939">#include <stdio.h> #include <math.h> int main() { double x=10.0; int y; for(y=0; y<3; y+=2) printf("%.2f\n", pow(x,y)); printf("y=%d", y); return 0; }</pre>	

ข้อ	คำถาม	คำตอบ
2.4	<p>แสดงผลลัพธ์ของโปรแกรม</p> <pre>#include <stdio.h> double x; void my_func(double a); int main() { x = 1.1; printf("x=%.2lf \n",x+1); my_func(x); x=x+10; printf("x=%.2lf \n", x); return 0; } void my_func(double a) { printf("x=%.2lf \n",a); printf("x=%.2lf\n",x); }</pre>	
2.5	<p>จงแสดงผลลัพธ์ของโปรแกรม โดยสมมติให้รับค่าจากผู้ใช้เป็น A, B และ exit ตามลำดับ</p> <pre>#include<string.h> #include<stdio.h> #define SL 100 int Left(void) { char S[SL]; int count=0; do { printf("string:\t"); scanf("%s",S); if(strcmp(S,"exit") != 0) ++count; }while(strcmp(S,"exit") != 0); return count; } int main() { printf("word count:%d",Left()); return 0; }</pre>	
2.6	<p>จากนิยามฟังก์ชัน nfib จงหาค่า nfib(5)</p> <pre>int nfib(int n) { if (n==0 n==1) return 1; else return 4+nfib(n-1)+nfib(n-2); }</pre>	

3. จงเติมช่องว่างในโปรแกรมให้สมบูรณ์ เพื่อหาค่า Power of Power ของจำนวนเต็ม m โดยถ้าค่า $n=0$ จะได้ค่าเป็น 1 และจะยกกำลังค่า m ตามจำนวน n ตัวอย่างเช่น ถ้า $n=1$ ค่าของ m จะมีค่าเท่ากับ m ถ้า $n=2$ ค่าของ m จะมีค่าเท่ากับ m^m ถ้า $n=3$ ค่าของ m จะมีค่าเท่ากับ $m^{(m^m)}$ ถ้า $n=4$ ค่าของ m จะมีค่าเท่ากับ $m^{(m^{(m^m)})}$ โดยโปรแกรมรับค่า m, n จากผู้ใช้แล้วเรียกใช้งานฟังก์ชัน (8 คะแนน)

```
#include<stdio.h>
int power(int r, int p) //3 คะแนน
{

}
int power_of_power(int m, int n) //3 คะแนน
{

}
void main() //2 คะแนน
{ int m,n;

}
}
```

***** จบตอนที่ 1 *****

ตอนที่ 2 (30 คะแนน, 15%, 60 นาที)

โครงสร้างข้อมูล (Structure)

1. จากโครงสร้างข้อมูลที่กำหนดให้ จงเขียนโปรแกรมเพื่อให้ทำงานตามคำสั่งในข้อ 1.1-1.10 (10 คะแนน)

```
struct article {
    char title[60]; // ชื่อสินค้า
    char cat[50]; // กลุ่มสินค้า
    int num; // จำนวน
    float price; // ราคา
};
```

- 1.1 ประกาศตัวแปรชื่อ art เป็นตัวแปรสตรัคเจอร์ article โดยให้มีค่าเริ่มต้นของสมาชิกเป็น "Mama" "Food" 20 และ 14.5 ตามลำดับ

- 1.2 กำหนดค่าสมาชิก title ของตัวแปร art ใหม่ให้มีชื่อสินค้าเป็น "Yummy"

- 1.3 ประกาศตัวแปรชื่อ art2 เป็นตัวแปรสตรัคเจอร์ article

- 1.4 รับค่าจาก keyboard เป็นข้อความและเก็บเป็นชื่อสินค้า title ของ art2

- 1.5 รับค่าจาก keyboard เป็นจำนวนเต็มเก็บเป็นจำนวนสินค้า num ของ art2

- 1.6 แสดงผลข้อมูลสมาชิกทุกตัวของ art2 ออกทางจอภาพ

- 1.7 ประกาศตัวแปร art3 เป็นตัวแปรอาร์เรย์ของสตรัคเจอร์แบบ article สามารถเก็บข้อมูลได้ 5 ชุด

- 1.8 กำหนดชื่อสินค้าชิ้นแรก title ของตัวแปร art3 ให้มีค่าเป็น "Mama"

- 1.9 กำหนดจำนวนสินค้าชิ้นสุดท้าย num ของตัวแปร art3 ให้มีค่าเท่ากับ 3

- 1.10 แสดงข้อมูลราคา price ของสินค้าทุกชิ้นในตัวแปร art3 ออกทางจอภาพ

2. จงตอบคำถามต่อไปนี้ (20 คะแนน)

2.1 จงนิยามโครงสร้างข้อมูลเป็นชนิดข้อมูลใหม่ชื่อ date มีสมาชิก ดังนี้ (2 คะแนน)

- วัน (day) เป็นตัวเลขจำนวนเต็ม
- เดือน (month) เป็นตัวเลขจำนวนเต็ม
- ปี (year) เป็นตัวเลขจำนวนเต็ม

2.2 จงนิยามโครงสร้างข้อมูลเป็นชนิดข้อมูลใหม่ชื่อ employee มีสมาชิกดังนี้ (2 คะแนน)

- ชื่อ (name) เป็นตัวอักษรไม่เกิน 20 ตัวอักษร
- วันเข้าทำงาน (start) เป็นชนิดข้อมูล date (ข้อ 2.1)
- เงินเดือน (salary) เป็นตัวเลขทศนิยม

2.3 จงเขียนฟังก์ชันเพื่อรับค่าข้อมูลจาก keyboard แล้วเก็บค่าไว้ในสมาชิกทุกตัวของตัวแปร e จำนวน n คน

(4 คะแนน)

ตัวอย่างผลลัพธ์

```

NAME#1 : Mana
START#1 : 12/12/2558
SALARY#1 : 15000
:
NAME#5 : Manee
START#5 : 11/11/2558
SALARY#5 : 15000
    
```


}

2.6 จงเรียกใช้ฟังก์ชัน input incsalary และ findemp ในฟังก์ชันหลักให้ถูกต้อง (4 คะแนน)

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

```
#define N 5
```

```
int main( )
```

```
{    employee emp[N];
```

```
    // ประกาศตัวแปรเพิ่มเติมตามความเหมาะสม
```

```
    // เรียกใช้ฟังก์ชัน input
```

```
    // เรียกใช้ฟังก์ชัน incsalary เพื่อเพิ่มเงินเดือนของพนักงานทุกคนๆ ละ 3.5%
```

```
    // เรียกใช้ฟังก์ชัน display
```

```
    display(emp);
```

```
    // เรียกใช้ฟังก์ชัน findemp และแสดงชื่อของพนักงานที่ทำงานนานที่สุด
```

```
    return 0;
```

```
}
```

```
/****** จบตอนที่ 2 *****/
```

ตอนที่ 3 (20 คะแนน, 10%, 50 นาที)

โปรแกรมประยุกต์

พิจารณาข้อมูลนักเทนนิสมืออาชีพ ประเภทชายเดี่ยวของ ATP World Tour

Rank (อันดับ)	PlayerID (รหัสประจำตัว)	Name (ชื่อ-นามสกุล)	Nationality (สัญชาติ)	Age (อายุ)	Points (คะแนน)	Updated (วันที่อัปเดตข้อมูล)
1	2005120	Novak Djokovic	Serbia	28	15285	18/11/2015
2	2006031	Andy Murray	England	28	8470	18/11/2015
3	2002013	Roger Federer	Switzerland	34	7340	18/11/2015
4	2008045	Stan Wawrinka	Switzerland	30	6500	16/11/2015
5	2004110	Rafael Nadal	Spain	29	4630	17/11/2015
6	2012044	Tomas Berdych	Czech Republic	21	4620	18/11/2015
7	2003102	David Ferrer	Spain	33	4305	16/11/2015
...

1. ให้กำหนดชนิดข้อมูล **Date** และ **Player** (โดยใช้ typedef) แบบโครงสร้างข้อมูล โดยโครงสร้างข้อมูล Date ประกอบด้วย day, month, year และ โครงสร้างข้อมูล Player ประกอบด้วย playerId, name, nationality, age, points และ updated ให้กำหนดชนิดข้อมูลอย่างเหมาะสม (4 คะแนน)

2. จงเขียนนิยามของฟังก์ชันสำหรับรับค่าข้อมูลของผู้เล่น ประกอบด้วย playerId, name, nationality, และage และบันทึกค่าเก็บไว้ในอาร์เรย์ P ฟังก์ชันส่งค่าคืนกลับเป็นจำนวนผู้เล่นที่ได้กรอกข้อมูล (4 คะแนน)

ต้นแบบของฟังก์ชัน `int inputPlayers(Player P[]);`

ตัวอย่างการทำงานของฟังก์ชัน เช่น (ตัวอักษรเอนคือข้อมูลที่รับจากผู้ใช้)

Input Player Information

Number of players: **10**

Player#1

PlayerID: **2003102**

Name: **David Ferrer**

Nationality: **Spain**

Age: **33**

Player#2

PlayerID: **2005120**

Name: **Novak Djokovic**

Nationality: **Serbia**

Age: **28**

... ใส่ข้อมูลไปเรื่อยๆจนครบ 10 คน ...

Player#10

PlayerID: **20122044**

Name: **Tomas Berdych**

Nationality: **Czech Republic**

Age: **21**

นิยามของฟังก์ชัน

3. จงเขียนนิยามของฟังก์ชันสำหรับ อัปเดตข้อมูลคะแนนของผู้เล่น โดยข้อมูลที่รับประกอบด้วย playerId, คะแนนใหม่ (points) และวันที่อัปเดต(updated) และบันทึกค่าเก็บไว้ในอาร์เรย์ P ให้ตรงกับข้อมูลของผู้เล่น playerId คนนั้นๆ พารามิเตอร์ n คือจำนวนข้อมูลผู้เล่นในอาร์เรย์ P (4 คะแนน)

ต้นแบบของฟังก์ชัน `void updatePoints (Player P[], int n);`

ตัวอย่างการทำงานของฟังก์ชัน เช่น

```
Update Points of Players
Number of players to update points: 2
Player#1
PlayerID: 2005120
Novak Djokovic, Current points = 15000
Enter new points: 15285
Updated date (dd/mm/yyyy): 18/11/2015
Player#2
PlayerID: 2003102
David Ferrer, Current points = 4000
Enter new points: 4305
Updated date (dd/mm/yyyy): 16/11/2015
```

นิยามของฟังก์ชัน

4. จงเขียนนิยามของฟังก์ชันสำหรับแสดงค่าข้อมูลรายละเอียดของผู้เล่น ตามลำดับที่เก็บในอาร์เรย์ P และจำนวนผู้เล่นกำหนดโดยพารามิเตอร์ n (4 คะแนน)

ต้นแบบของฟังก์ชัน void **showPlayers**(Player P[], int n);

ตัวอย่างการทำงานของฟังก์ชัน เช่น เมื่อ n = 5 (มีข้อมูลของผู้เล่น 5 คนในอาร์เรย์)

Show Player Information

#	PlayerID	Name	Nationality	Age	Points	Updated
1	2005120	Novak Djokovic	Serbia	28	15285	18/11/2015
2	2006031	Andy Murray	England	28	8470	18/11/2015
3	2002013	Roger Federer	Switzerland	34	7340	18/11/2015
4	2008045	Stan Wawrinka	Switzerland	30	6500	16/11/2015
5	2004110	Rafael Nadal	Spain	29	4630	17/11/2015

นิยามของฟังก์ชัน

5. จงเติมส่วนของโปรแกรมหลัก(main) ให้สมบูรณ์ สมมติว่ามีการนิยามสตรัคเจอร์ และฟังก์ชันต่างๆ เรียบร้อยแล้ว สามารถเรียกใช้ฟังก์ชันต่างๆ ใน main ได้เลย โดยโปรแกรมหลักมีลำดับการทำงานดังนี้
- ประกาศตัวแปรอาร์เรย์ที่เก็บข้อมูลผู้เล่นเทนนิส (ความยาวสูงสุดไม่เกิน 500 คน)
 - เรียกใช้ฟังก์ชัน `inputPlayers()` เพื่อให้ผู้ใช้กรอกข้อมูลผู้เล่น
 - เรียกใช้ฟังก์ชัน `updatePoints()` เพื่อใส่ข้อมูลคะแนนของผู้เล่น
 - เรียกใช้ฟังก์ชัน `ranking()` เพื่อจัดลำดับผู้เล่น n คน ตามคะแนน โดยฟังก์ชัน `ranking()` มีโปรโตไทป์ดังนี้ `void ranking (Player P[],int n);`
 - เรียกใช้ฟังก์ชัน `showPlayers()` เพื่อแสดงข้อมูลของผู้เล่นในอาร์เรย์ที่จัดลำดับเรียบร้อยแล้ว (4 คะแนน)

โปรแกรมหลัก

```
#include <stdio.h>
//ส่วนการประกาศสตรัคเจอร์ และ ฟังก์ชันต่างๆ
//ฟังก์ชัน inputPlayers, updatePoints, showPlayers, ranking
//สมมติสตรัคเจอร์และฟังก์ชันประกาศเรียบร้อยแล้ว

int main()
{ //เติมโปรแกรมหลักให้สมบูรณ์
```