

PRINCE OF SONGKLA UNIVERSITY
FACULTY OF ENGINEERING



Midterm Examination: ภาคการศึกษาที่ 1/2559

Date: 11 ตุลาคม พ.ศ. 2559 09.00-11.00 น.

Subject Number: 242-207

Room: S201, S817

Subject: Fundamental Programming I

ทฤษฎีในการสอบ มีโทษขั้นต่ำ คือ ปรับตกในรายวิชาที่ทฤษฎี และพักการเรียน 2 ภาคการศึกษา

อ่านรายละเอียดของข้อสอบ และคำแนะนำให้เข้าใจก่อนเริ่มทำข้อสอบ

รายละเอียดของข้อสอบ:

รหัสนักศึกษา

เวลา 2 ชั่วโมง (120 คะแนน: 120 นาที)

ชื่อ-สกุล

เอกสารมีทั้งหมด 6 หน้า (ไม่รวมหน้านี้)

สำหรับอาจารย์

คำถามจำนวน 4 ข้อ

สิ่งที่สามารถนำเข้าห้องสอบได้:

อนุญาต: เครื่องเขียน

ไม่อนุญาต: หนังสือ และเครื่องคิดเลข

คำแนะนำ:

- เขียนชื่อ (ไม่ต้องเขียนนามสกุล) และรหัสนักศึกษา ในทุกหน้า
- พยายามทำทุกข้อ และคำตอบทั้งหมดให้ทำในข้อสอบชุดนี้
- คำตอบส่วนใดอ่านไม่ออก จะถือว่าคำตอบนั้นผิด
- อ่านคำสั่งในแต่ละข้อให้ชัดเจนว่า เขียนโปรแกรมบางส่วน เขียนฟังก์ชัน หรือเขียนทั้งโปรแกรม รวมไปถึงข้อกำหนดเพิ่มเติม และหมายเหตุในข้อนั้นๆ
- การเขียนโปรแกรมในแต่ละข้อ อาจจะไม่ต้องเขียนตามคำสั่งย่อยทั้งหมด แต่คะแนนจะลดลงตามส่วน
- การเขียน code จะต้องตั้งชื่อตัวแปรให้เหมาะสม และมี comment ในจุดสำคัญต่างๆ โดยให้ทั้งหมดเป็นไปตามหลักการเขียนโปรแกรมที่ดี

ข้อ	คะแนน
1	
2	
3	
4	

อ.สุธน: ผู้ออกข้อสอบ

ข้อที่ 1 ความรู้พื้นฐาน

(25 คะแนน: 25 นาที)

1.1 จงเขียนฟังก์ชัน ที่เหมาะสมสำหรับการหาผลบวกของสมาชิกเลขจำนวนเต็มในอาร์เรย์ (5 คะแนน)

.....

.....

.....

.....

1.2 จากข้อมูลที่กำหนดให้ จงเขียนคำสั่งตามที่กำหนดในแต่ละข้อย่อย (5 คะแนน)

```
int a[5] = {3, 2, 7, -4, 6};
int *ptr;
```

ให้ ptr ช่างอิงไปยังสมาชิกที่มีหมายเลข index = 2

ให้ ptr เปลี่ยนค่าตัวถัดไปจากที่ ptr ชี้อู่ให้เป็น 0

(ห้ามใช้เครื่องหมาย [])

1.3 จงเขียนโค้ดเพื่อจองหน่วยความจำด้วยเทคนิค Dynamic Memory Allocation เทียบเท่ากับโค้ดข้างล่าง

```
int d[5];
```

(5 คะแนน)

1.4 จงเขียนแผนภาพแสดงค่าของตัวแปร และการเชื่อมโยงของตัวแปรทุกตัวในหน่วยความจำ เมื่อโปรแกรมทำงานจนสิ้นสุดบรรทัดสุดท้ายของโค้ดข้างล่าง (10 คะแนน)

```
int a = 7, *p, b[5] = {2, 1, 1, 3, 5}, *s[2];
s[0] = &a;
s[1] = &b[2];
*(s[1] + 2) = 6;
p = b + 3;
(*p)++;
```

คำตอบข้อ 1.4

ข้อที่ 2 ข้อความและตัวอักษร

(35 คะแนน : 35 นาที)

2.1 จงเขียนฟังก์ชันที่ทำการตรวจสอบข้อความ s1 และ s2 เพื่อค้นหาตัวอักษรตัวแรกที่แตกต่าง ทั้งนี้ให้ถือว่าข้อความ s1 และ s2 มีความยาวเท่ากันเสมอ (20 คะแนน)

เช่น ข้อความ s1 คือ Hello และข้อความ s2 เป็น Henry ผลการทำงานของฟังก์ชันจะคืนค่าเป็น pointer ไปยังอักษร l ตัวแรกของข้อความ s1

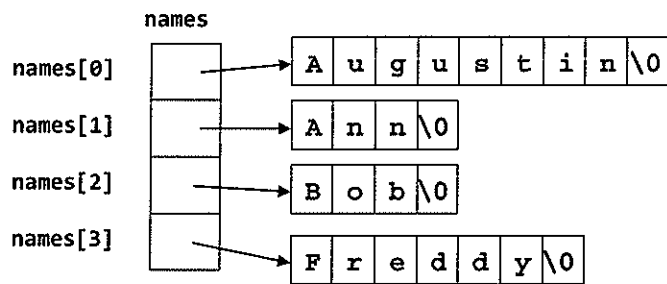
```
char *strdiff ( char *s1, char *s2 );
//คืนค่า pointer ไปยังตัวอักษรที่แตกต่าง ในข้อความ s1
```

ทั้งนี้ นักศึกษาจะต้องไม่ใช่ [] ในการเขียนโปรแกรม

```
char *str_diff ( char *s1, char *s2 ) {
```

```
}
```

2.2 จงเขียนส่วนของโปรแกรมเพื่อรับคำศัพท์ภาษาอังกฤษ (ข้อความไม่เกิน 127 ตัวอักษร) ตามรูปแบบที่แสดงไว้ในภาพข้างล่าง (15 คะแนน)



ข้อที่ 3 การจัดการหน่วยความจำ / โครงสร้างข้อมูล

(35 คะแนน: 35 นาที)

โปรแกรมรับข้อมูลของ Path ซึ่งจะต้องรับจำนวนจุดที่เป็นส่วนประกอบของ Path (ใน ตย. Path มีจุด 3 จุดเป็นส่วนประกอบ) แล้วรับข้อมูลจุดของ Path ที่ละจุดตามลำดับ หลังจากนั้นโปรแกรมทำการคำนวณหาระยะทางของ Path ซึ่งจะเป็นค่าผลรวมของระยะห่างแต่ละจุดใน Path จากจุดแรกไปยังจุดสุดท้าย

จงเติมส่วนที่ขาดหายไป โดยใช้ฟังก์ชัน และโครงสร้างข้อมูลต่างๆ ต้องถูกใช้อย่างเหมาะสม

```

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

typedef struct {
    int x, y;
} Point;

typedef struct {
    int amount;
    Point *points;
} Path;


... function prototypes ...

int main(){
    Path p;
    path_input(&p);
    printf("Path Distance = %.2f\n", path_distance(&p));
    path_clear(&p);
    return 0;
}

void path_input(Path *p){ //***** (15 marks)
    int i, amount;
    printf("Point amount: ");

    for(i = 0; i < amount; i++){
        printf(" %2d. ", i + 1);
    }
}

```



The diagram shows a path consisting of three points (represented by black dots) connected by straight lines. The points are arranged in a V-shape, with one point at the top and two points below it, one to the left and one to the right. Lines connect the top point to each of the bottom points.

```
void point_input(Point *p){ //***** (5 marks)
    printf("(x,y): ");

}

void path_clear(Path *p){ //***** (5 marks)
    //free unused memory

}

float path_distance(Path *p){ //***** (10 marks)
    int i;
    float total = 0.0;

    return total;
}

float line_distance(Point *p1, Point *p2){
    return sqrt(pow(p1->x - p2->x, 2) +
                pow(p1->y - p2->y, 2));
}
```

ข้อที่ 4 ไฟล์

(25 คะแนน: 25 นาที)

นายดำขำกำลังติดเกมออนไลน์ RPG ตัวหนึ่งอย่างมาก โดยเกมนี้สามารถสร้างตัวละครหลักได้หลายตัว นายดำขำทราบมาว่า ไฟล์เซฟเกมที่บันทึกข้อมูลของตัวละคร คือไฟล์ save.bin โดยเพื่อนของนายดำขำแจ้งว่า ไฟล์นี้ไม่ได้มีการเข้ารหัส และเก็บข้อมูลเป็นแบบไฟล์ไบนารีธรรมดา

ไฟล์ save.bin เก็บข้อมูลของตัวละครหลักแต่ละตัวตามโครงสร้างของ Character ที่แสดงในโค้ดด้านล่าง จงเขียนโค้ดเพื่อช่วยนายดำขำเปลี่ยนแปลง max_hp ของตัวละครหลักทุกตัวเป็น 9999

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

typedef struct {
    char name[128];
    ...
    int max_hp;
} Character;

int main(){
    int num, i, done = 0;
    Character c;

    FILE *fp =          ;

    fclose(fp);
    return 0;
}
```

```
size_t fread(void *ptr, size_t sz, size_t n, FILE *fp)
size_t fwrite(const void *ptr, size_t sz, size_t n, File *fp);
int fseek ( FILE * stream, long int offset, int origin );
```