

PRINCE OF SONGKLA UNIVERSITY
FACULTY OF ENGINEERING



Midterm Examination: ภาคการศึกษาที่ 1/2553

Date: 3 สิงหาคม พ.ศ. 2553

Subject Number: 241-207

Room: A401, S101, S203

Subject Title: Data Structures and Computer Programming Techniques

ทฤษฎีในการสอบ มีโทษขั้นต่ำ คือ ปรับตกในรายวิชาที่ทฤษฎี และพักการเรียน 1 ภาคการศึกษา

อ่านรายละเอียดของข้อสอบ และคำแนะนำให้เข้าใจก่อนเริ่มทำข้อสอบ

รหัสนักศึกษา

รายละเอียดของข้อสอบ:

สำหรับอาจารย์

เวลา 3 ชั่วโมง (180 คะแนน: 180 นาที)

เอกสารมีทั้งหมด 7 หน้า (ไม่รวมหน้านี้)

คำถามจำนวน 4 ข้อ + ข้อพิเศษ 1 ข้อ

สิ่งที่สามารถนำเข้าห้องสอบได้:

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	

ชื่อ รหัสนักศึกษา

ข้อที่ 1 ความรู้พื้นฐาน

(80 คะแนน)

1.1 จากอาร์เรย์ a จงเขียนส่วนของโค้ด เพื่อนับจำนวนตัวเลขที่เป็นเลขคู่

(10 คะแนน)

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

.....

.....

.....

.....

.....

1.2 ประกาศตัวแปรพอยเตอร์ ptr และใช้พอยเตอร์ ptr เปลี่ยนค่าของ x ให้เป็น 10.0

(10 คะแนน)

```
double x = 0;
```

.....

.....

.....

1.3 จงเขียนส่วนของโค้ดที่ใช้พอยเตอร์ cptr และ Dereferencing แทน [] ที่ให้ผลแบบเดียวกันกับโค้ดข้างล่าง

(10 คะแนน)

```
int i; //char *cptr;  
char str[] = "Hello";  
for(i = 0; str[i] != '\0'; i++)  
    if(str[i] >= 'a' && str[i] <= 'z')  
        str[i] += ('A' - 'a');
```

ชื่อ

รหัสนักศึกษา

.....
.....
.....
.....
.....
.....

1.4 จากโค้ดที่กำหนดให้ จงเขียนส่วนของโค้ด เพื่อใช้พอยเตอร์ ptr กำหนดค่าให้กับ e โดยให้ name มีค่าเป็น "John Doe" และมีค่าเงินเดือนเป็น 25000 บาท (10 คะแนน)

```
typedef struct {  
    char name[128];  
    double salary;  
} Employee;  
Employee e;  
Employee *ptr;
```

.....
.....
.....
.....
.....
.....

1.5 จงเขียนแผนภาพแสดงค่าของตัวแปร และการเชื่อมโยงของตัวแปรทุกตัวในหน่วยความจำ เมื่อโปรแกรมทำงานจนถึงจุดบรรทัดสุดท้ายของโค้ดข้างล่าง (20 คะแนน)

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

1.6 จงเลือกคำตอบ A-I ที่สัมพันธ์กับโจทย์มากที่สุด โดยคำตอบหนึ่งอาจถูกใช้ตอบในโจทย์มากกว่า 1 ข้อ (20 คะแนน)

- | | |
|--|--|
| A. อยู่ติดกันในหน่วยความจำ | F. มีการสำเนาค่าของตัวแปร |
| B. ตัวแปรที่ประกาศถูกทำลายเมื่อเสร็จสิ้นการทำงาน | G. ปิดท้ายด้วย '\0' |
| C. เก็บค่าตำแหน่งในหน่วยความจำของตัวแปรอื่น | H. ไม่สามารถนำไปใช้แสดงผลบนหน้าจอได้ |
| D. Constant Pointer | I. ผลของการบวกบนขึ้นอยู่กับชนิดของตัวแปรที่อ้างอิงอยู่ |
| E. Dereferencing | |

- | | |
|----------------------------|--------------------------|
| 1.6.1 คุณสมบัติของอาเรย์ | (0).....(1)..... |
| 1.6.2 คุณสมบัติของฟังก์ชัน | (2).....(3)..... |
| 1.6.3 คุณสมบัติของพอยเตอร์ | (4).....(5).....(6)..... |
| 1.6.4 คุณสมบัติของสตริง | (7).....(8).....(9)..... |

ข้อที่ 2 ข้อความและตัวอักษร

(40 คะแนน)

จงเขียนฟังก์ชันเพื่อทำการตรวจสอบว่า ข้อความ s2 เป็นส่วนหนึ่งของข้อความ s1 หรือไม่

```
int is_part(char *s1, char *s2);
```

- เช่น "Hello" และ "orl" และ "o" เป็นส่วนหนึ่งของข้อความ "Hello World"
- ฟังก์ชันทำงานแบบ case sensitive เช่น "wo" ไม่เป็นส่วนหนึ่งของ "Hello World"
- ถ้า s2 มีความยาวมากกว่า s1 ให้ถือว่า s2 ไม่เป็นส่วนหนึ่งของ s1
- ถ้า s2 ข้อความเหมือนกันกับ s1 ให้ถือว่า s2 เป็นส่วนหนึ่งของ s1
- ฟังก์ชันคืนค่า 1 หาก s2 เป็นส่วนหนึ่งของ s2 แต่จะคืน 0 ในกรณีตรงกันข้าม
- ฟังก์ชันนี้จะต้องมีการเรียกใช้ คือ strcmp และ strlen เท่านั้น และไม่อนุญาตให้เรียกใช้ฟังก์ชันอื่นอีก

```
int strcmp (const char * str1, const char * str2, size_t num);  
size_t strlen(const char *s);
```

ข้อที่ 3 อาร์เรย์ พอยเตอร์และหน่วยความจำ

(30 คะแนน)

โค้ดข้างล่าง เป็นโค้ดที่แสดงการประมวลผลกับอาร์เรย์ data จงใช้ข้อมูลในโค้ดตอบคำถามข้อ 3.1 และ 3.2

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

#define LEN 5

int* foo(int data[], int len){
    int *r = (int*) malloc(sizeof(int) * len);
    int *p = r;
    int i;
    for(i = 0; i < len; i++, p+=2){
        *p = data[i]/2;
        *(p + 1) = data[i] - *p;
    }
    return r;
}

int main(){
    int a[5] = {1, 4, 3, 0, 9};
    int i;
    int *r = foo(a, LEN);

    for(i = 0; i < LEN * 2; i++){
        printf("%d ", r[i]);
    }
    printf("\n");

    return 0;
}
```

3.1 โค้ดข้างต้นมีความผิดพลาดในการจัดการหน่วยความจำหรือไม่ หากมีให้แก้ไข หรือเพิ่มเติม ให้เขียนไว้ร่วมกับโค้ดด้านบน พร้อมคำอธิบายประกอบ

3.2 จงเขียนผลลัพธ์ที่ถูกแสดงเมื่อรันโปรแกรมข้างต้น (หลังจากแก้ไขข้อผิดพลาดเกี่ยวกับการจัดการหน่วยความจำแล้ว)

.....

.....

ชื่อ

รหัสนักศึกษา

ข้อที่ 4 โครงสร้างข้อมูล

(30 คะแนน)

จากโค้ดของที่กำหนดให้ จงเขียนฟังก์ชันที่ขาดหายไป โดยมีข้อกำหนดเพิ่มเติมว่า ฟังก์ชัน `total_distance` ต้องเรียกใช้ฟังก์ชัน `distance` อย่างเหมาะสม

```
float distance(Point *p1, Point *p2){
    return sqrt(pow(p1->x - p2->x, 2) +
               pow(p1->y - p2->y, 2));
}
int main(){
    Point p[5];
    int i;
    for(i = 0; i < 5; i++){
        printf("P%02d:", i + 1);
        input_point(&p[i]);
    }
    printf("=%.2f", total_distance(p, 5));
    return 0;
}
```

รูปแบบการแสดงผล

```
P01: <0 0>
P02: <3 4>
P03: <3 4>
P04: <0 0>
P05: <5 5>
= 17.07
```

โค้ดข้างต้นจะรับพิกัดจุด x, y จำนวน 5 จุด แล้วหาระยะทางการเดินทางจากจุดที่ 1 ไป 2 + ระยะทางจาก 2 ไป 3 ไปจนถึงจากจุดที่ 4 ไปจุดที่ 5

ชื่อ รหัสนักศึกษา

ข้อที่ 5 ข้อพิเศษ

(0 คะแนน)

***นศ. ไม่จำเป็นต้องทำข้อนี้ โดยเลือกทำด้วยความสมัครใจ หากตอบมาจะได้รับการตรวจและแจ้งผลเป็น Good-Pass-Fail

จงเขียนโปรแกรมโดยสมบูรณ์ เพื่อรับข้อมูลดังนี้ คือ จำนวนของตัวเลขที่ต้องการประมวลผล ตามด้วยข้อมูลตัวเลขตามจำนวนที่ได้ระบุไว้ หลังจากนั้นทำการคำนวณเพื่อตัดข้อมูลที่มากที่สุดออก และตัวข้อมูลที่น้อยที่สุดออก ก่อนนำมาหาค่าเฉลี่ยของข้อมูล โดยให้มีการใช้งานฟังก์ชันดังต่อไปนี้

```
int *input_datas(int len); //return pointer to datas
int *find_max(int *d, int len); //return pointer to max_data
int *find_min(int *d, int len); //return pointer to min_data
int remove_data(int *t, int *d, int len); //return new length of datas
float find_average(int *d, int len); //return average of data
```

ทั้งนี้ให้ทำการวิเคราะห์ว่า พารามิเตอร์แต่ละตัวควรถูกใช้อย่างไร และฟังก์ชันควรจะถูกเรียกใช้อย่างไร

ห้ามใช้เครื่องหมาย [] หรือรูปเหมือน *(p + i) ในโค้ดข้อนี้ไม่ว่าในกรณีใดๆ

(นศ.สามารถใช้ด้านหลังของข้อ 4 เพื่อตอบคำถามข้อนี้ด้วย)