

PRINCE OF SONGKLA UNIVERSITY
FACULTY OF ENGINEERING



Final Examination: ภาคการศึกษาที่ 1/2557

Date: 15 ธันวาคม พ.ศ. 2557 (13.30-15.30 น.)

Subject Number: 241-207, 242-207

Room: A401, R200, S203, S817

Subject: Data Structures and Computer Programming Techniques, Fundamental Programming I

ทฤษฎีในการสอบ มีโทษขั้นต่ำ คือ ปรับตกในรายวิชาที่ทฤษฎี และพักการเรียน ๑ ภาคการศึกษา

อ่านรายละเอียดของข้อสอบ และคำแนะนำให้เข้าใจก่อนเริ่มทำข้อสอบ

รายละเอียดของข้อสอบ:

รหัสนักศึกษา

เวลา 2 ชั่วโมง (120 คะแนน: 120 นาที)

ชื่อ-สกุล

เอกสารมีทั้งหมด 7 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 อาจจะไม่จำเป็นต้องเขียนตามคำสั่งย่อยทั้งหมด แต่คะแนนจะลดลงตามส่วน
- การเขียน code จะต้องตั้งชื่อตัวแปรให้เหมาะสม และมี comment ในจุดสำคัญต่างๆ โดยให้ทั้งหมดเป็นไปตามหลักการเขียนโปรแกรมที่ดี

ข้อ	คะแนน
1	
2	
3	
4	

อ.สุธน: ผู้ออกข้อสอบ

ข้อที่ 1 Warm Up

(30 คะแนน)

1.1 จงเลือกโครงสร้างข้อมูลที่เหมาะสมกับโปรแกรมประยุกต์แต่ละข้อ โดยโครงสร้างข้อมูลที่มีให้เลือกใช้ มีดังนี้

- a) Array with dynamic memory allocation
- b) Linked List
- c) Hash Table
- d) Binary Search Tree

(6 คะแนน)

ลักษณะการทำงาน	คำตอบ (เลือกที่ถูกที่สุด)
ผู้ใช้ระบุจำนวนของข้อมูลเมื่อรันโปรแกรม เน้นการอ่านข้อมูลเพียงอย่างเดียว	
รับข้อมูลจากผู้ใช้ โดยไม่จำกัดจำนวน เน้นการอ่านข้อมูล โดยต้องการประสิทธิภาพในการค้นหาข้อมูลสูงสุด	
เน้นการแทรกและลบข้อมูล	

1.2 จากโครงสร้างข้อมูลที่กำหนด จงเขียนนิยามของฟังก์ชัน `modify_data` เพื่อเปลี่ยนค่าที่ถูกบรรจุไว้ในโหนดที่

กำหนด

(6 คะแนน)

```
struct listnode
{
    int data;
    struct listnode *next;
};
typedef struct listnode LN;
void modify_data(LN* node, int new_value);
```

.....

.....

.....

.....

1.3 จากฟังก์ชัน `mktree` ที่กำหนดให้ จงเรียกใช้ฟังก์ชันดังกล่าว เพื่อสร้างหรือตั้งรูปข้างล่าง

(9 คะแนน)

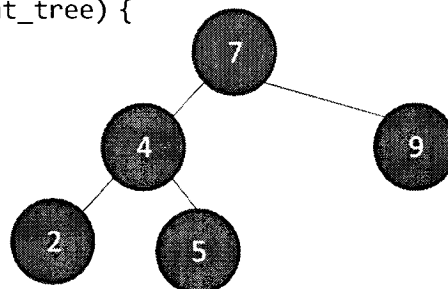
```

TREE mk_tree(int x, TREE left_tree, TREE right_tree) {
    TREE temp;

    temp = (TREE) malloc(sizeof(TREENODE));
    temp->data = x;
    temp->left = left_tree;
    temp->right = right_tree;

    return temp;
}

```



.....

.....

.....

.....

.....

.....

.....

1.4 จากกราฟที่กำหนดให้ จงเขียนโครงสร้างข้อมูลแทนกราฟ โดยใช้เทคนิค Adjacency Matrix และ Adjacency Lists (เขียนทั้งสองอย่าง)

(9 คะแนน)



ข้อที่ 2 Linked List

(30 คะแนน)

2.1 จงเขียนแผนภาพแสดงหน่วยความจำ (ต้องแสดงว่าพอยน์เตอร์แต่ละตัวชี้อยู่ที่ใด และค่าของข้อมูลแต่ละจุดเป็นเท่าใด) เมื่อมีการ execute ชุดคำสั่งที่กำหนดไว้ในแต่ละบล็อกเสร็จแล้ว (เป็นคำตอบต่อเนื่อง) โดยข้อมูลที่ใช้ทำการป้อนให้กับโปรแกรมเป็นตัวเลขตามลำดับดังนี้ 5 2 3 (10 คะแนน)

หมายเหตุ โครงสร้างของ LN เป็นไปตามที่กำหนดในข้อ 1.2

```
int i, data;
LNP head, tail, tmp;
head = tail = (LN*)malloc(sizeof(LN));
head->data = 0;
head->next = NULL;
```

แผนภาพ

```
for(i = 0; i < 3; i++){
    scanf("%d", &data);
    tmp = (LN*)malloc(sizeof(LN));
    tmp->data = data;
    tmp->next = NULL;
    tail->next = tmp;
    tail = tmp;
}
head->next->next->data = 7;
```

แผนภาพ

2.2 โปรแกรมหนึ่ง ต้องการฟังก์ชันที่ใช้สร้าง linked-list ที่ข้อมูลใน linked-list จะเรียงลำดับจากน้อยไปหามาก จงเขียนฟังก์ชัน `insert_in_order` เพื่อแทรกข้อมูลที่กำหนดให้ในตำแหน่งที่จะรักษาลำดับของข้อมูลให้ถูกต้อง

โครงสร้างของ List Node และ ดันแบบฟังก์ชัน `insert_in_order` เป็นดังนี้ (20 คะแนน)

```
struct listnode {
    int data;
    struct listnode *next;
};
typedef listnode LN;

void insert_in_order(LN *head, int data);
```

คำตอบ

ข้อที่ 3 More Structures**(25 คะแนน)**

3.1 จงอธิบายให้ชัดเจนว่า เราสามารถนำ stack มาใช้ในการแปลงเลขฐาน 10 เป็นเลขฐาน 2 ได้อย่างไร
(ใช้ภาพประกอบ) (10 คะแนน)

.....

.....

.....

.....

.....

.....

.....

3.2 จงอธิบายถึงสาเหตุที่เราต้อง mark ใน Hash table ว่า ณ.ตำแหน่งนั้น เคยมีข้อมูล แม้ว่าเราลบข้อมูล
ออกไปแล้ว (ใช้ภาพประกอบ) (15 คะแนน)

.....

.....

.....

.....

.....

ข้อที่ 4 Tree

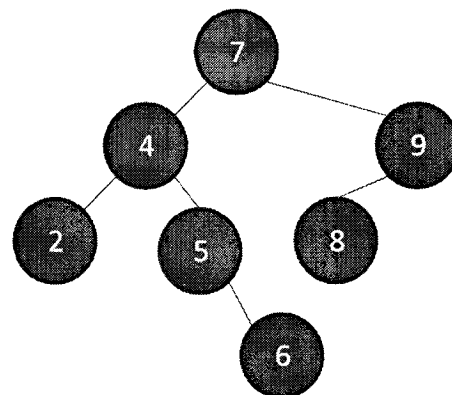
(35 คะแนน)

จาก BST ที่กำหนดให้ ตอบคำถามข้อ 4.1-4.2

4.1 จงตอบคำถามเกี่ยวกับคุณลักษณะของ Tree ตามที่กำหนดให้

(5 คะแนน)

คุณลักษณะ	คำตอบ
Root	
Children ของ 4	
Descendants ของ 4	
Leaf	
ความสูง	



4.2 จงแสดงข้อมูลในรูปแบบ pre-order, in-order และ post-order

(9 คะแนน)

..... Pre-order:

..... In-order:

..... Post-order:

4.3 จงอธิบายถึงคุณลักษณะของ BST และระบุข้อดีของ BST โดยให้กล่าวถึง Big O Notation ด้วย

(6 คะแนน)

.....

.....

.....

.....

.....

.....

4.4 จากโครงสร้างข้อมูลที่กำหนดให้

(15 คะแนน)

```
struct treenode{
    struct treenode *left;
    int data;
    struct treenode *right;
};
typedef struct treenode TREENODE;
typedef TREENODE *TREE;
```

จงเขียนฟังก์ชัน `count_leaf` เพื่อหาจำนวนของ leaf ในทรี
