PRINCE OF SONGKLA UNIVERSITY

FACULTY OF ENGINEERING

Final Examination: ภาคการศึกษาที่ 1/2556 **Date:** 7 กันยายน พ.ศ. 2556

Subject Number: 241-207, 242-207 **Room:** S201, A401, Robot

Time : 13.30 - 15.30

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

ทุจริตในการสอบ มีโทษขั้นต่ำ คือ ปรับตกในรายวิชาที่ทุจริต และพักการเรียน 1 ภาคการศึกษา

อ่านรายละเอียดของข้อสอบ และคำแนะนำให้เข้าใจก่อนเริ่มทำข้อสอบ

รายละเอียดของข้อสอบ:

เวลา 2 ชั่วโมง (120 คะแนน: 120 นาที)

เอกสารมีทั้งหมด 7 หน้า (ไม่รวมหน้านี้)

คำถามจำนวน 3 ข้อ

สิ่งที่สามารถนำเข้าห้องสอบได้:

อนุญาต: เครื่องเขียน

ไม่อนุญาต: หนังสือ และเครื่องคิดเลข

รหัสนักศึกษา .....

ชื่อ-สกุล .....

สำหรับอาจารย์

ข้อ	คะแนน
1	
2	
3	

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

อ.สุธน: ผู้ออกข้อสอบ

ข้อที่ 1 องค์ความรู้ (50 คะแนน)

1.1 จงระบุว่า เราสามารถใส่ข้อมูลที่มีผลลัพธ์ของการ Hash เหมือนกันลงไปใน Hash Table ได้หรือไม่ จงอธิบายโดยใช้ภาพประกอบ

.....

.....

.....

.....

1.2 จงสร้าง `struct` ชื่อ `intdl` ที่สามารถนำไปใช้ในการสร้าง Doubly Linked List ที่แต่ละโหนดสามารถเก็บตัวเลขได้หนึ่งตัว

.....

.....

.....

.....

.....

1.3 จากโครงสร้างข้อมูลที่กำหนด จงเขียนนิยามของฟังก์ชัน **modify_data** เพื่อเปลี่ยนค่าที่ถูกบรรจุไว้ในโหนดที่กำหนด

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

.....

.....

.....

.....

.....

1.4 จงเลือกโครงสร้างข้อมูล a, b, c, d ที่เหมาะสมกับโปรแกรมประยุกต์แต่ละข้อในตารางข้างล่าง

```
a) Array with static memory allocation
b) Array with dynamic memory allocation
c) Linked List
d) Linked List of Array
```

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

1.5 จงอธิบายให้ชัดเจนว่า เราสามารถนำ stack มาใช้ในการคำนวนค่าของนิพจน์แบบ Postfix ได้อย่างไร

.....

.....

.....

.....

.....

.....

.....

.....

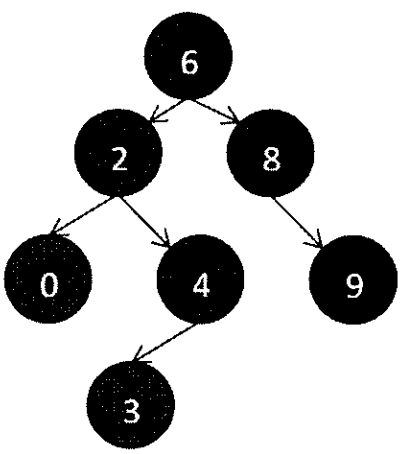
.....

ข้อที่ 2 Tree (30 คะแนน)

จาก BST ที่กำหนดให้ ตอบคำถามข้อ 2.1-2.2

2.1 จงตอบคำถามเกี่ยวกับคุณลักษณะของ Tree ตามที่กำหนดให้ (10 คะแนน)

คุณลักษณะ	คำตอบ
Root	
Children ของ 2	
Descendants ของ 2	
Leaf	
ความสูง	



2.2 จงแสดงข้อมูลในแบบ pre-order, in-order และ post-order (10 คะแนน)

Pre-order: .....

In-order: .....

Post-order: .....

2.3 จากโครงสร้างข้อมูลที่กำหนดให้ (10 คะแนน)

```
struct treenode{
   struct treenode *leftptr;
   int data;
   struct treenode *rightptr;
};
typedef struct treenode TREENODE;
typedef TREENODE *TREE;
```

จงเขียนฟังก์ชัน `validate_bst` เพื่อทำการตรวจสอบว่าทรีเป็น BST หรือไม่ หากเป็นให้คืนค่า 1 หากไม่เป็นให้คืนค่า 0

ข้อที่ 3 ประยุกต์ใช้ (40 คะแนน)

จงเขียนนิยามของฟังก์ชันที่ขาดหาย สำหรับโปรแกรมเพื่อประมวลผลข้อมูลทางเคมี จะได้มีการอ่านข้อมูลจากไฟล์ data.txt เพื่อสร้าง Linked List โดยแต่ละโหนดประกอบด้วย ชื่อทางเคมีซึ่งเป็นตัวอักษร 3 ตัว และ จำนวนหน่วย ทั้งนี้หากในไฟล์พบชื่อทางเคมีที่ได้สร้างเป็นโหนดไว้แล้ว ก็ให้ใช้โหนดเดิมได้ โดยทำการบวกจำนวนหน่วยใหม่เข้าไปกับจำนวนที่มีอยู่เดิม

```
ATA 3
RCG 1
ATA 2
KTY 1
...
```

เช่น ในข้อมูลตัวอย่าง 4 บรรทัดแรก จะทำให้เกิดโหนด 3 โหนดใน Linked List คือ โหนดของ ATA, RCG และ KTY โดยมีจำนวนหน่วยเป็น 5, 1 และ 1 ตามลำดับ เมื่อได้ Linked List แล้วจะทำการประมวลข้อมูลดังนี้

- ตัดโหนดทุกโหนดสำหรับชื่อทางเคมี ที่มีจำนวนหน่วยมากกว่า 10 (threshold)
- แสดงผลต่าง **ระหว่าง** ผลรวมจำนวนหน่วยของ ชื่อทางเคมี ATA, RMA และ ZTA **และ** ผลรวมจำนวนหน่วยของ ชื่อทางเคมี KTY และ BGA โดยหากชื่อทางเคมีนั้นไม่ปรากฏใน Linked List ก็ให้ถือว่าจำนวนหน่วยเป็น 0

หมายเหตุ การให้คะแนนจะให้ตามความเหมาะสมของการเขียนโค้ด ทั้งนี้อาจมีการนิยามฟังก์ชันเพิ่มเติม ตามความจำเป็น

```
#include <stdio.h>
#define A_NUM 3
#define B_NUM 2
struct listnode {
   char name[4];
   int unit;
   struct listnode *next;
};
typedef struct listnode LN;

void load_data(char* filename, LN **hptr);
LNP threshold_find(LN *head, int threshold);
void delete_node(LN **hptr, LN *node);
int sum_nodes(LN *head, char *givens[], int len);
void append_node(LN *hptr, LN **tptr, char *name, int unit);
```

```
int main()
{
    LNP head = NULL;
    char *set_a[A_NUM] = {"ATA", "RMA", "ZTA"};
    char *set_b[B_NUM] = {"KTY, BGA"};
    LNP tmp;
    int diff;
    load_data("data.txt", &head);
    while((tmp = threshold_find(head, 10)) != NULL)
        delete_node(&head, tmp);

    diff = sum_nodes(head, set_a, A_NUM) - sum_nodes(head, set_b, B_NUM);

    printf("Diff = %d\n", diff);
    return 0;
}

void append_node(LN *hptr, LN **tptr, char *name, int unit){
    LNP tmp = (LNP)malloc(sizeof(ListNode));
    strcpy(tmp->name, name);
    tmp->unit = unit;
    tmp->next = NULL;

    if(*hptr == NULL){
        *hptr = tmp;
    }else{
        (*tptr)->next = tmp;
    }
    *tptr = tmp;
}

void load_data(char* filename, LN **hptr){
    LNP tail = NULL;
    FILE *fp = fopen("data.txt", "r");
    char name[4];
    int unit;
    while(!feof(fp)){
        fscanf(fp, "%s %d\n", name, &unit);
        append_node(hptr, &tail, name, unit);
    }
    fclose(fp);
}
```