

FACULTY OF ENGINEERING
CHULALONGKORN UNIVERSITY
2110211 Introduction to Data Structures

YEAR II, Second Semester, Mid-term Examination, December 22, 2000, Time 8:30-11:30

หมายเหตุ

1. ไม่อนุญาตให้นำตำราต่างๆ หรือเครื่องคำนวณใดๆ เข้าห้องสอบ
2. ข้อสอบมีทั้งหมด 12 ข้อ 5 หน้า
3. ให้เขียนคำตอบลงในช่องว่างที่เว้นไว้ในกระดาษคำถามนี้ เท่านั้น

ชื่อ-สกุล..... เลขประจำตัว..... ตอนเรียนที่ CR58.....

1. (3 คะแนน) เรียงลำดับฟังก์ชันต่อไปนี้ตาม growth rate (Big-O ของฟังก์ชันนั้น) จากน้อยไปมาก โดยเขียนเลขลำดับลงในช่องว่าง ฟังก์ชันที่มี growth rate เท่ากันให้มีเลขลำดับเดียวกัน

ฟังก์ชัน	N log N	N + log N	N log log N	2 ^{log N}	N log N ²	2N
ลำดับ						

2. (3 คะแนน) ในการคำนวณหาคำตอบของ postfix expression abc+d*ef*++ โดยใช้ stack จงวาด stack ในขณะที่มี element อยู่ใน stack เป็นจำนวนมากที่สุด

3. (4 คะแนน) เขียน method printTreeInverse เพื่อพิมพ์ข้อมูลใน binary search tree จากมากไปน้อย กำหนดให้ element ของ BinaryNode เป็น String

```
public void printTreeInverse( ) {
    printTreeInverse( root );
}
private void printTreeInverse( BinaryNode t ) {

}
}
```

4. (4 คะแนน) Sorted Linked List เป็น Linked List ที่เก็บข้อมูลที่เรียงลำดับจากน้อยไปมาก กำหนดให้มี class ชื่อ SortedLinkedList อยู่แล้ว ซึ่งคล้ายกับ class LinkedList ที่เราได้ศึกษากันในชั้นเรียน จงเขียน method เพื่อ insert ข้อมูลใหม่เข้าไป โดยข้อมูลใน list เรียงลำดับอย่างถูกต้อง (เขียนตอบในแผ่นถัดไป)

```
class ListNode
{
    Comparable element;
    ListNode next;
    . . .
}
public class SortedLinkedList {
```

```
private ListNode header;
public void insert ( Comparable x )
{
}
}
```

5. (4 คะแนน) กำหนดให้มี AVL tree เริ่มต้นที่ไม่มี node เลย (empty tree) แล้วมีการ insert ตัวเลขตามลำดับต่อไปนี้คือ 1, 2, 3, 8, และ 6 เข้าไปใน tree นี้ จงวาด AVL tree นี้แต่ละครั้งหลังจากที่มีการ insert
6. คุณบานบุรีอยากทราบว่าตนเองสืบเชื้อสายมาจากคนดังในประวัติศาสตร์หรือไม่ จึงจ้างนักสืบให้หาข้อมูลเพื่อมาสร้างเป็นต้นไม้ โดยมีคุณบานบุรีเป็น root node และมีคุณพ่อเป็น left child คุณแม่เป็น right child เช่นนี้เรื่อย ๆ
- 6.1. (2 คะแนน) ทำไมต้นไม้นี้เป็น binary tree และต้นไม้เป็น binary search tree หรือไม่
- 6.2. (2 คะแนน) เขียน class ของ node สำหรับต้นไม้
- 6.3. (2 คะแนน) ถ้าคุณบานบุรีแต่งงานแล้วมีลูก ให้แสดงการเปลี่ยนแปลงของต้นไม้ ก่อนและหลังสภาพการณ์ทำไม่จึงเป็นเช่นนั้น

- 6.4. (4 คะแนน) แสดง method สำหรับการแต่งงานแล้วมีลูกของคุณบานบุรี ให้แสดงเป็น pseudo code ใน method ได้
- 6.5. (2 คะแนน) ถ้าคุณพ่อคุณแม่ของคุณบานบุรีมีลูกหลายคน ต้นไม้จะเปลี่ยนไปหรือไม่อย่างไร ให้เหตุผลประกอบ
- 6.6. (4 คะแนน) หากต้องการเก็บข้อมูลเกี่ยวกับการเป็นบุตรบุญธรรมในต้นไม้ด้วย ต้นไม้จะเปลี่ยนไปอย่างไร
- 6.7. (4 คะแนน) ถ้าบรรพบุรุษสองคน สืบเชื้อสายมาจากบรรพบุรุษคนเดียวกัน จะทำอย่างไร แสดงภาพประกอบคำอธิบาย
7. (4 คะแนน) ให้วิเคราะห์ Big-Oh (running time) สำหรับส่วนของโปรแกรมต่อไปนี้
- ```
sum = 0;
for(i = 0; i < n; i++)
 for(j = 0; j < n * n; j++)
 sum++;
```

8. (4 คะแนน) จงเขียน method contains เพิ่มให้ class StackAr ซึ่งทำหน้าที่หาดูว่ามี object ที่กำหนดให้อยู่ใน stack หรือไม่ ถ้ามีก็คืน true ถ้าไม่มีก็คืน false

```
public class StackAr {
 private Object [] theArray;
 private int topOfStack;
 ...
 public boolean contains(Object x) {

 }
}
```

9. (4 คะแนน) method isBST ทำหน้าที่ตรวจสอบให้แน่ใจว่าต้นไม้ที่รับเข้ามานั้นเก็บข้อมูลตรงตามกฎของ binary search tree หรือไม่ ถ้าเป็น binary search tree ก็คืน true มิฉะนั้นก็คืน false  
จงเติมคำสั่งต่างๆ ในช่องที่เว้นว่างไว้ เพื่อให้ isBST ทำงานถูกต้อง

```
class BinaryNode {
 Comparable element;
 BinaryNode left, right;
}

public class BinarySearchTree {
 BinaryNode root;
 ...

 public boolean isBST(BinarySearchTree t) {
 boolean b1, b2;

 if (t == null) return true;
 b1 = b2 = true;
 if (t.left != null) {

 }
 if (t.right != null) {

 }
 return b1 & b2;
 }
}
```

1

2

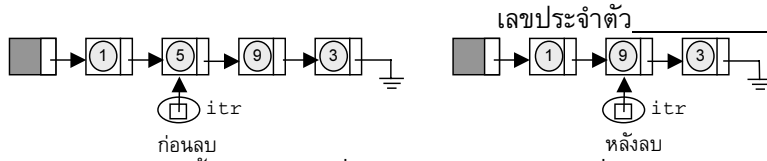
10. (4 คะแนน) ใน class LinkedList ที่เราได้ศึกษากันในชั้นเรียนนั้น method remove ซึ่งมีหัว method แสดงข้างล่างนี้

```
public void remove(Object x)
```

ใช้เวลาการทำงานเป็น  $O(n)$  โดยที่  $n$  เป็นจำนวนข้อมูลใน linked list เพราะที่ต้องวิ่งหา node ที่อยู่ก่อน node ที่เก็บ  $x$  ให้พบก่อนแล้วจึงเปลี่ยน link ถ้าเราเพิ่ม remove อีกแบบที่มีหัว method ข้างล่างนี้

```
public void remove(LinkedListItr itr)
```

โดยทำหน้าที่ลบข้อมูลใน list ที่อ้างอิงถึงด้วย itr แสดงดังรูปต่อไปนี้



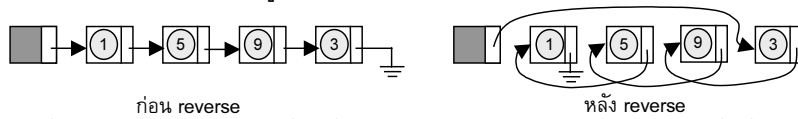
จงเขียนรายละเอียดของ method นี้ให้สมบูรณ์ เพื่อให้สามารถทำงานตามที่ต้องการได้สำเร็จในเวลาคงตัว (นั่นคือ  $O(1)$ ) ทั้งนี้ให้ถือว่า

- itr ไม่ได้อ้างอิงข้อมูลตัวสุดท้ายใน list
- ก่อนลบ list นี้มีข้อมูลอย่างน้อยสองตัว
- ไม่มี iterator อื่นใด ที่อ้างอิง nodes อื่นๆ ใน list ยกเว้น itr

```
public void remove(LinkedListItr itr) {
}

```

11. (4 คะแนน) นายอภิสิทธิ์เขียน method reverse เพิ่มเข้าไปใน class LinkedList เพื่อทำหน้าที่กลับลำดับของ nodes ต่างๆ ใน list ดังตัวอย่างในรูปข้างล่างนี้



เขียนๆ ไป ชักไม่มั่นใจ ก็เลยเว้นว่างส่วนที่ไม่มั่นใจให้เติมกัน จงเติมคำสั่งจำนวนหนึ่งที่เว้นว่างไว้ข้างล่างนี้เพื่อให้ reverse ทำงานถูกต้อง (คำแนะนำ : คำสั่งที่หายไปมีเพียงสองคำสั่ง)

```
public void reverse() {
 header.next = reverse(header.next);
}
private ListNode reverse(ListNode p) {
 if (p == null) return p;
 if (p.next == null) return p;
 ListNode r = reverse(p.next);

 return r;
}

```

12. (3 คะแนน) เราสามารถสร้าง binary tree ได้ด้วย cursors ซึ่งมีลักษณะคล้ายกับการใช้ cursor สร้าง linked list จงวาดรูปต้นไม้ที่รากอยู่ที่ slot 5 ในตารางข้างล่างนี้

| slot | element | left | right |
|------|---------|------|-------|
| 0    | -       | 12   | 0     |
| 1    | E       | 4    | 5     |
| 2    | D       | 0    | 0     |
| 3    | I       | 1    | 12    |
| 4    | X       | 11   | 0     |
| 5    | A       | 6    | 9     |
| 6    | H       | 13   | 0     |
| 7    | B       | 0    | 0     |
| 8    | G       | 0    | 0     |
| 9    | F       | 10   | 8     |
| 10   | W       | 0    | 2     |
| 11   | J       | 0    | 10    |
| 12   | K       | 3    | 2     |
| 13   | C       | 7    | 0     |