รหัสนักศึกษา_____

1. Big O vs. Benchmark

1.1 [2 คะแนน] ประสิทธิภาพใดที่สามารถวัดผลได้ด้วย Benchmark แต่ไม่สามารถวัดผลได้ด้วยการคำนวณ Big O

1.2 [2 คะแนน] ประสิทธิภาพใดที่สามารถวัดผลได้ด้วยการคำนวณ Big O แต่ไม่สามารถวัดผลได้ด้วย Benchmark

2. การคำนวณ Big O

2.1 [2 คะแนน] หากนักศึกษาต้องการคำนวณ Big O ของขั้นตอนวิธีในการรวบรวมจำนวนกระโปรงนักศึกษาที่ขายได้ในเดือนมีนาคม นักศึกษาจะใช้อะไรเป็นขนาดของปัญหา (n)

2.2 [2 คะแนน] สำหรับขั้นตอนวิธีที่ใช้เวลาเป็น $O[(\log n)^2]$ หากขนาดของปัญหาเพิ่มขึ้นเป็น 100 เท่า (จาก n เป็น 100n) แล้ว ขั้นตอนวิธีนี้จะใช้เวลาเพิ่มขึ้นเป็นกี่เท่า

2.3 [2 คะแนน] สำหรับขั้นตอนวิธีที่ใช้เวลาเป็น $O(n\ log\ n)$ หากขั้นตอนวิธีนี้ใช้เวลา 100ms เมื่อ n=10 แล้ว ขั้นตอนวิธีนี้จะใช้เวลาประมาณเท่าไหร่เมื่อ n=1000

รหัสนักศึกษา_____

3. จากโปรแกรมหา $\lfloor n^2 \rfloor$ ต่อไปนี้ (จำนวนเต็มที่มากที่สุดที่น้อยกว่ากำลังสองของ n)

```
int floorSqrN1 (double n) {
  int r = 1;
  while(r<n*n) {
    r++;
  }
  return r-1;
}
```

```
int floorSqrN2(double n) {
  int r = (int)(n*n);
  while(r<n*n) {
    r++;
  }
  r-1;
}
```

3.1 [5 คะแนน] ให้นับจำนวนคำสั่งของกรณีที่แย่ที่สุดของ floorSqrN1 และสรุปว่า floorSqrN1 มี Big O เป็นเท่าใด

3.2 [5 คะแนน] ให้นับจำนวนคำสั่งของกรณีที่แย่ที่สุดของ floorSqrN2และสรุปว่า floorSqrN2มี Big O เป็นเท่าใด

รหัสนักศึกษา_____

4. [6 คะแนน] ให้เขียนโปรแกรมเพื่อให้ b เป็น กำลังสองของกลับทาง (reverse) ของ a

ตัวอย่าง หาก a = {5, 4, 3, 2, 1} จะได้ b = {1, 4, 9, 16, 25}

5. [6 คะแนน] ให้เขียนโปรแกรมเพื่อกลับทางราย (reverse) การโยง (linked list)

ตัวอย่าง จาก head→[1]→[2]→[3]→[4]→null เปลี่ยนเป็น head→[4]→[3]→[2]→[1]→null

รหัสนักศึกษา_____

6. จากส่วนของโปรแกรมต่อไปนี้

```
int index = 0;
for(int i=1; i<a.length; i++) {
        if(a[i]>a[index]) index = i;
}
int tmp =a[index];
a[0] = a[index];
a[index] = tmp;
System.out.println(a[0]);
```

[5 คะแนน] 6.1 อธิบายการทำงานของส่วนของโปรแกรมนี้ พร้อมทั้งแสดงผลของโปรแกรม

[5 คะแนน] 6.2 ให้นับจำนวนคำสั่งของกรณีที่แย่ที่สุดส่วนของโปรแกรมนี้ พร้อมสรุปว่า Big O เป็นเท่าใด

รหัสนักศึกษา_____

7. จากโปรแกรมย่อยต่อไปนี้ (กำหนด class Node ของ linked list)

```
void method7(Node p) {
        if( p==null ) return false;
        boolean result = false;
        while(p.next!=null) {
                if(p.data<p.next.data) {
                        int tmp = p.data;
                        p.data = p.next.data;
                        p.next.data = tmp;
                        result = true;
                }
                p = p.next;
        }
        return result;
}
```

7.1 [4 คะแนน] หากกำหนดให้ linked list มีข้อมูลเป็น head → [3] → [1] → [5] → [4] → null แล้ว หลังจากการเรียก method7(head) โครงสร้าง linked list จะกลายเป็นอย่างไร

7.2 [4 คะแนน] Big O ของโปรแกรมย่อยนี้เป็นเท่าไร

7.3 [4 คะแนน] หากเรียกใช้ method7 ตามส่วนของโปรแกรมด้านล่างแล้ว ผลของ linked list จะออกมาเป็นอย่างไร

```
while(method7(head));
```

รหัสนักศึกษา_____

8.1 [2 คะแนน] การดำเนินการใดบ้างของสแตกที่ต้องใช้เวลาเป็น O(1) เสมอ

8.2 [5 คะแนน] กำหนดให้มีสแตกว่างเปล่าอยู่ 2 สแตก ชื่อว่า stackA และ stackB ให้นักศึกษาเขียนข้อมูลภายในสแตก (ให้ข้อมูลขวาสุดเป็นด้านบนของสแตก) ที่ผ่านการดำเนินการต่อไปนี้

8.2.1 stackA.push(5) stackA: stackB:	8.2.2 stackB.push(1) stackA: stackB:
8.2.3 stackA.push(2) stackA: stackB:	8.2.4 stackB.push(stackA.pop()) stackA: stackB:
8.2.5 stackA.push(stackB.pop()-stackA.pop()) stackA: stackB:	8.2.6 stackA.push(1) stackA: stackB:
8.2.7 stackB.push(4) stackA: stackB:	8.2.8 stackA.push(stackB.pop()*stackB.pop()) stackA: stackB:
8.2.9 stackA.push(stackA.pop()) stackA: stackB:	8.2.10 stackB.push(stackA.top()+1) stackA: stackB:

รหัสนักศึกษา_____

9.1 [2 คะแนน] การดำเนินการใดบ้างของ queue ที่ต้องใช้เวลาเป็น O(1) เสมอ

9.2 [5 คะแนน] กำหนดให้มี queue ว่างเปล่าอยู่ 2 queue ชื่อว่า qA และ qB ให้นักศึกษาเขียนข้อมูลภายใน queue (ให้ข้อมูลซ้ายสุดเป็นด้านหน้า queue) ที่ผ่านการดำเนินการต่อไปนี้

9.2.1 qA.enqueue(6) qA: qB:	10.2 qB.enqueue(2) qA: qB:
9.2.3 qA.enqueue(1) qA: qB:	9.2.4 qB.enqueue(qA.dequeue()) qA: qB:
9.2.5 qA.enqueue(4) qA: qB:	9.2.6 qB.enqueue(7) qA: qB:
9.2.7 qA.enqueue(qA.dequeue()*qB.dequeue()) qA: qB:	9.2.8 qB.enqueue(qA.dequeue()*qA.dequeue()) qA: qB:
9.2.9 qA.enqueue(qA.dequeue()+2) qA: qB:	9.2.10 qB.enqueue(qB.dequeue()-5) qA: qB:

รหัสนักศึกษา_____

10. จากนิพจน์ทางคณิตศาสตร์ต่อไปนี้ 6 * ((2 + 3) * (5 – 4 / 2)) ให้วาดข้อมูลใน stack และ queue

10.1 [3 คะแนน] หลังจากดำเนินการด้วย Shunting Yard Algorithm ผ่านเลข 3 แล้ว

10.2 [3 คะแนน] หลังจากดำเนินการด้วย Shunting Yard Algorithm ผ่านเครื่องหมายลบ [–] แล้ว

รหัสนักศึกษา_____

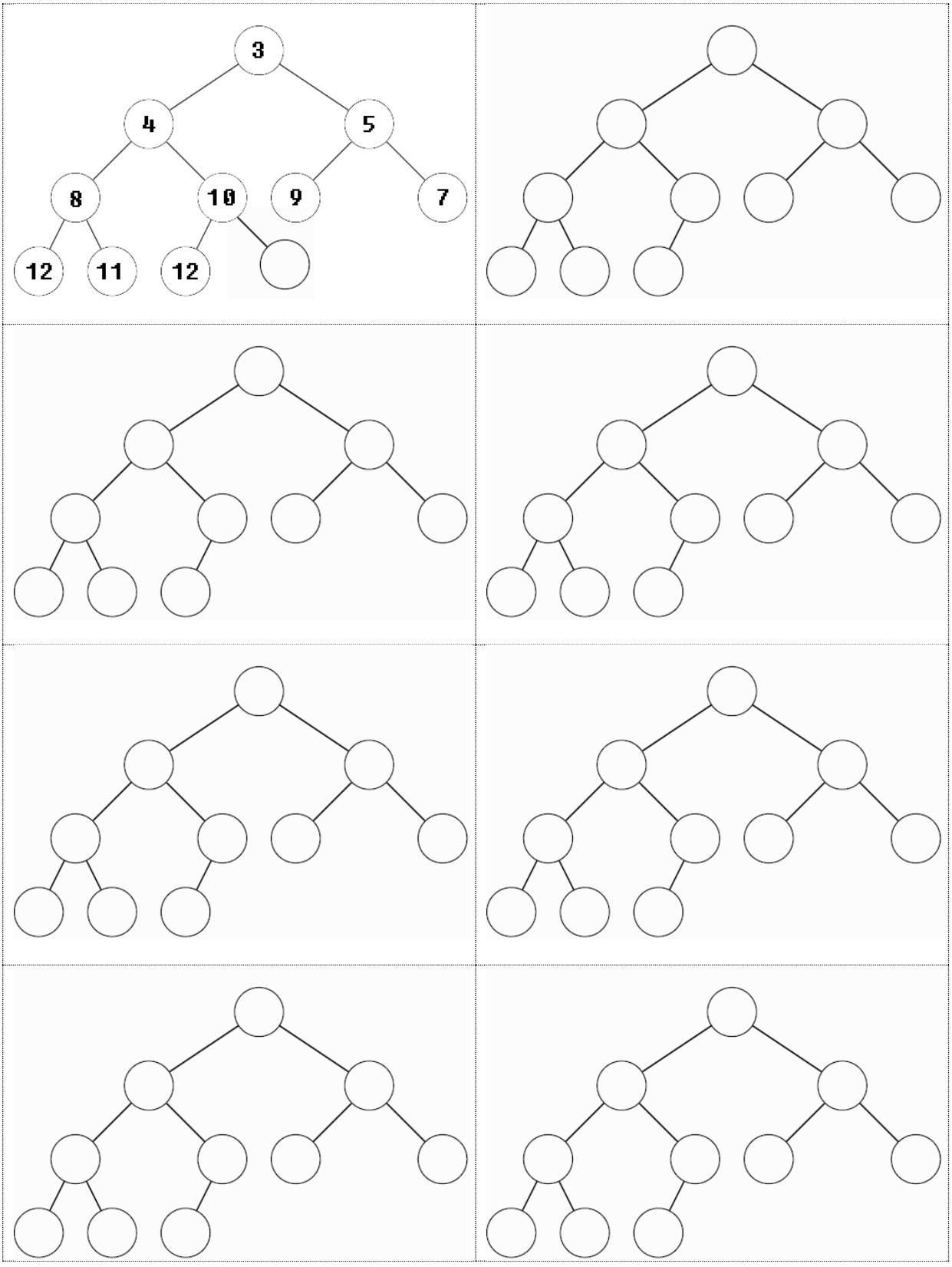
11. จากนิพจน์ Reversed Polish Notation นี้ 5 9 7 - 4 * 2 / 2 - * ให้วาดข้อมูลใน stack

11.1 [3 คะแนน] หลังจาก RPN Evaluation Algorithm ดำเนินการผ่านเลข 4 แล้ว

11.2 [3 คะแนน] หลังจาก RPN Evaluation Algorithm ดำเนินการผ่านตัวดำเนินการ - ตัวที่สองแล้ว

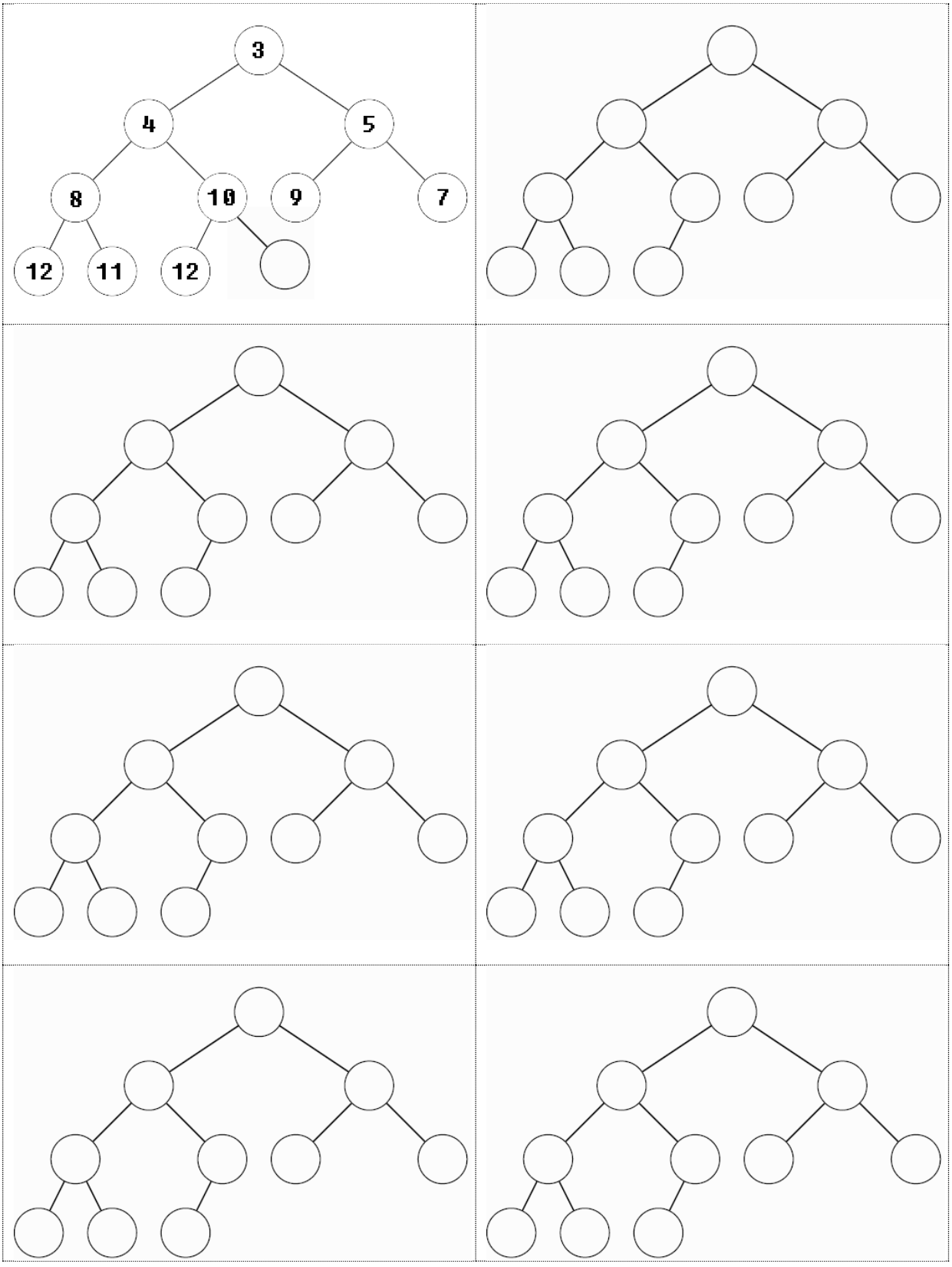
รหัสนักศึกษา_____

12. [4 คะแนน] จงเขียนลำดับขั้นในการ enqueue(6) ลงบน heap นี้



รหัสนักศึกษา_____

13. [4 คะแนน] จงเขียนลำดับขั้นในการ dequeue() จาก heap นี้



รหัสนักศึกษา_____

14. [10 คะแนน] กำหนดให้ A และ B เป็นอาร์เรย์ของตัวเลขที่ไม่มีการเรียงลำดับ ให้เขียนโปรแกรมเพื่อตรวจสอบว่า A เป็นเซ็ตย่อย (subset) ของ B หรือไม่ (สมาชิกทุกตัวของ A ต้องอยู่ในอาร์เรย์ B แต่อาจมีสมาชิกบางตัวใน B ไม่อยู่ใน A) และวิเคราะห์ว่ามีค่า Big O เป็นเท่าใด

<u>คำแนะนำ 1</u> เขียนเป็น pseudo code หรือ flow chart และวิเคราะห์ สามารถได้ 6/10 คะแนน

<u>คำแนะนำ 2</u> ใช้ heap (priority queue) (ไม่ต้องเขียน implementation ของ heap/priority queue)