

FACULTY OF ENGINEERING
CHULALONGKORN UNIVERSITY
2110210 PROG METH

Year 2nd, First Semester, Midterm Examination August 01, 2007. Time 13:00-16:00

ชื่อ-นามสกุล.....เลขประจำตัว.....ตอนเรียนที่.....เลขที่นั่ง CR58.....

หมายเหตุ

1. ข้อสอบมีทั้งหมด 4 หน้า รวมทั้งแผ่นนี้ด้วย คะแนนรวมทั้งหมด 25 คะแนน
2. อนุญาตให้นำคำราและเอกสารต่างๆ เข้าในห้องสอบได้ ดูเอกสารอิเล็กทรอนิกส์ที่เตรียมมาเองได้
3. เอกสารอิเล็กทรอนิกส์ ต้องก๊อปปี้ใส่ c:\temp ภายใน 15 นาทีแรกของการสอบ อินเทอร์เน็ตและเน็ตเวิร์คไครฟ์จะใช้ไม่ได้หลัง 15 นาทีแรก ดังนั้นจึงเตรียมเครื่องให้พร้อม
4. ไม่อนุญาตให้ดูเอกสารของเพื่อน ไม่ว่าจะเป็เอกสารธรรมดาหรืออิเล็กทรอนิกส์
5. ห้ามการหยิบยืมสิ่งใดๆ ทั้งสิ้นจากผู้สอบอื่นๆ เว้นแต่ผู้คุมสอบจะหยิบยืมให้
6. ห้ามการติดต่อสื่อสารใดๆ ทั้งสิ้น นอกจากคุยกับอาจารย์และ TA เท่านั้น คนที่ผู้คุมสอบเห็นกำลัง chat หรือเข้าเว็บบอร์ดคุยกับเพื่อน หรือแม้แต่มี msn อยู่ที่ task bar จะถือว่าทุจริตทันที แม้ว่าจะไม่ได้ sign in ก็ตาม
7. คนที่มื thumb drive เสียบอยู่ที่เครื่องจะถือว่าทุจริตทันที
8. มีไฟล์ให้ import ใ้ project อยู่ในโฟลเดอร์เดียวกับโจทย์นี้ ให้ import ก่อนอ่านโจทย์ทุกคน
9. ผู้ที่ประสงค์จะออกจากห้องสอบก่อนหมดเวลาสอบ สามารถออกได้แต่ต้องก่อนไม่น้อยกว่า 45 นาที
10. เมื่อหมดเวลาสอบ ผู้เข้าสอบต้องหยุดการเขียนหรือพิมพ์ใดๆ ทั้งสิ้น ให้ผู้เข้าสอบวางคีย์บอร์ดไว้บนตัวเครื่องแล้วปิดจอคอมพิวเตอร์ เจ้าหน้าที่คุมสอบจะก๊อปปี้งานจากเครื่องทีละคน
11. ห้ามผู้เข้าสอบลุกจากที่นั่ง จนกว่าเจ้าหน้าที่คุมสอบจะได้ก๊อปปี้งานของผู้เข้าสอบจากเครื่องไปแล้ว
12. ผู้ที่ปฏิบัติเข้าข่ายทุจริตในการสอบ ตามประกาศคณะวิศวกรรมศาสตร์ มีโทษคือ ได้รับสัญลักษณ์ F ในรายวิชาที่ทุจริต และพักการศึกษาอย่างน้อย 1 ภาคการศึกษา

รับทราบ

ลงชื่อนิสิต(.....)

อ่านโจทย์ให้หมดก่อน แล้วจึงเริ่มทำ

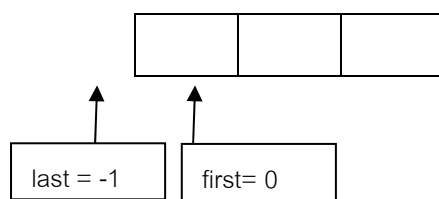
อาจารย์อยากให้นาย A นาย B และคุณ เขียน โค้ดของ queue ซึ่งเป็นการจำลองการเข้าต่อแถวเข้าคิว โดยอาจารย์ให้อินเตอร์เฟซมาดังไฟล์ Queue.java รายละเอียดของอินเตอร์เฟซนั้นเป็นดังนี้

```
public interface Queue {  
  
    // เป็นเมธอดที่ใช้ใส่ x เข้าไปต่อท้ายคิว  
    public void enqueue(Object x);  
  
    // เป็นเมธอดที่เอาสมาชิกออกจากหัวคิวไป เมธอดนี้รีเทิร์นสมาชิกตัวที่เอาออกนั้น  
    public Object dequeue();  
  
    // เป็นเมธอดที่รีเทิร์นสมาชิกตัวที่ k ของคิว (เริ่มนับจากตัวที่อยู่ตัวแรกของคิวเป็นตัวที่ 0)  
    public Object kthPosition(int k);  
}
```

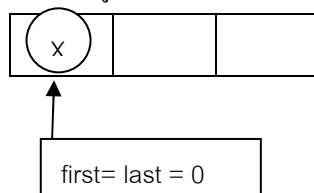
คุณกับนาย A และนาย B ได้ตกลงกันว่าจะใช้อาร์เรย์ในการทำคิวนี้ หลักการของการใช้อาร์เรย์จำลองการเข้าคิวของพวกคุณคือให้มี

- ค่า first บอกตำแหน่งของสมาชิกตัวแรกในคิว (ไม่จำเป็นจะต้องเป็นสมาชิกที่ index 0 ของอาร์เรย์นะ)
- ค่า last บอกตำแหน่งของสมาชิกตัวสุดท้ายในคิว (ไม่จำเป็นจะต้องเป็นสมาชิกตัวท้ายอาร์เรย์นะ)
- ค่า size บอกจำนวนสมาชิกในคิว ณ เวลาปัจจุบัน

เริ่มแรก เมื่อสร้างคิวขึ้นนั้นคุณให้มันมีอาร์เรย์ขนาด 3 ช่อง ดังรูป ซึ่งคิวที่สร้างใหม่นี้จะถือเป็นคิวเปล่า ไม่มีอะไรข้างใน

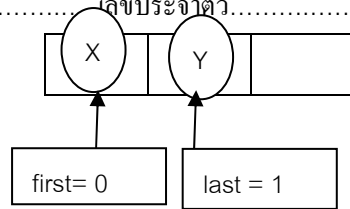


เมื่อทำการ enqueue X เข้าไป สถานะจะเปลี่ยนไปดังรูป

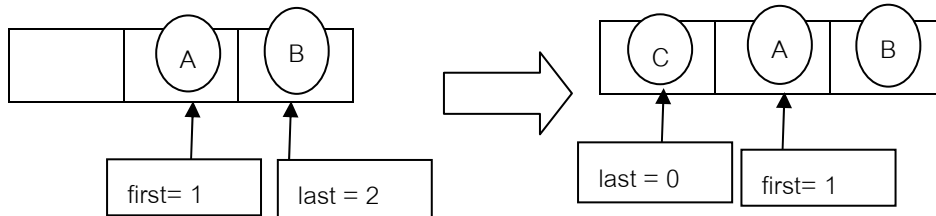


นั่นคือ การ enqueue นั้น last จะถูกเลื่อนไป แล้ว สมาชิกตัวใหม่จะถูกใส่ในตำแหน่งที่เลื่อนไปนั้น ดังนั้นถ้า enqueue Y เข้าไปต่อ เราก็จะได้อาร์เรย์ดังรูปข้างล่าง

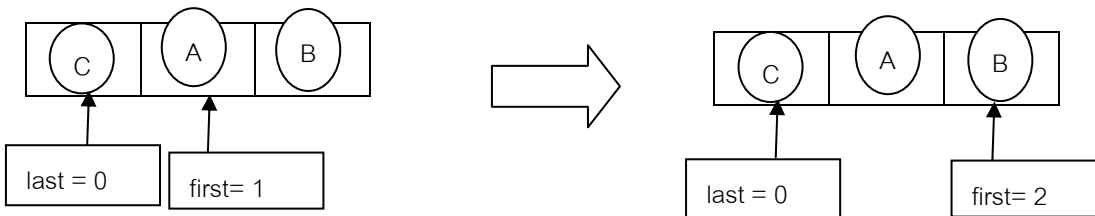
ชื่อ-นามสกุล..... เลขประจำตัว..... ตอนเรียนที่..... เลขที่ในCR58.....



แต่ถ้า คิวยังไม่เต็มและ last อยู่ที่อาร์เรย์อยู่แล้ว (เป็นไปได้อุปการ dequeue แล้วจะเข้าใจ) จะเกิดการ วนย้อนมา เพื่อใส่สมาชิกในตอนต้นของอาร์เรย์ ดังรูป เป็นการใส่ c เข้าไปในคิวที่มี A กับ B อยู่แล้ว



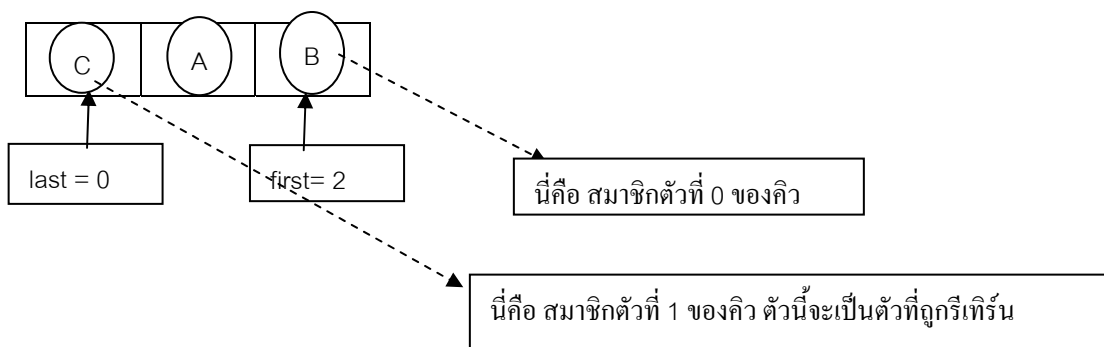
ส่วนการ dequeue นั้นจะเป็นการเอาของออกจากหัวคิว ซึ่งเป็นการเลื่อน first ไปหนึ่งช่องนั่นเอง รูปข้างล่างนี้ เป็นการเอาสมาชิกตัวแรก (ในที่นี้คือ A) ออกจากคิว



ซึ่งการเลื่อน first นั้น มีการวนอาร์เรย์ได้เช่นเดียวกับการเลื่อน last (ข้อสังเกต: A ยังอยู่ในอาร์เรย์นะ แต่ว่าไม่อยู่แล้วในฐานะสมาชิกของคิวนี้) อย่างลึ้มว่า A ต้องถูกรีเทิร์นจากเมธอด dequeue ด้วย

ส่วนการ dequeue คิวที่ไม่มีของอยู่ข้างใน จะได้คำตอบเป็น null

ในส่วนของ kthPosition นั้น ถ้า k เป็นตำแหน่งที่เป็นลบหรือเป็นตำแหน่งที่เกินกว่าจำนวนสมาชิกในคิว ให้รีเทิร์น null ส่วนถ้าเป็นตำแหน่งของสมาชิกในคิว ให้รีเทิร์นสมาชิกตัวนั้น รูปข้างล่างนี้แสดงผลการหาสมาชิกตัวที่ 1 ของคิว (เริ่มนับตัวแรกเป็นตัวที่ 0)



ชื่อ-นามสกุล.....เลขประจำตัว.....ตอนเรียนที่.....เลขที่ในCR58.....

นาย A ได้อาสาเขียนโค้ดขึ้นให้ก่อน โดยได้เขียนไฟล์ QueueArray.java มา ส่วนนาย B ก็เขียนโค้ด ของ TestQueueArray.java มา รวมทั้งเขียน ThePeople.java เพื่อจำลองคนเข้าคิวในการ test ด้วย

คุณ ในฐานะหัวหน้าทีม ได้เห็นโค้ดของเพื่อนทั้งสองคน พบว่า โค้ดของนาย A นั้นมีที่ผิดอยู่มาก ส่วนโค้ดการ test ของนาย B ก็ไม่ค่อยสมบูรณ์ คุณต้องแก้ไขโค้ดเสร็จสมบูรณ์ดังนี้

- โค้ดของนาย A ต้องสามารถทำตามนิยามที่คุยกันไว้ได้ข้างบนทั้งหมด ทุกเมธอดที่นาย A ให้มาต้องทำงานได้ ถูกต้อง (8 คะแนน)
- โค้ดของนาย B ต้องคอมไพล์ผ่าน (2 คะแนน)
- ให้การ enqueue เมื่ออาร์เรย์เต็ม จะต้องมีการจัดการด้วย โดยขยายอาร์เรย์ให้มีขนาดเป็นสองเท่า แล้วจัดพวก index ต่างๆใหม่ ให้เรียบร้อย ก่อนจะเดิมสมาชิกลงไป ให้แก้ไขโค้ดของทั้งนาย A และนาย B ให้รองรับตรงนี้ได้ (7 คะแนน)
- ในโค้ดของนาย B ให้แก้ไข testIsFull ให้ตรวจการที่เราเติมคิวเต็มไปสามที (3 คะแนน)
- ให้เขียน javadoc ของคลาส QueueArray ให้ครบทุกฟิลด์และเมธอด แล้ว สร้างตัวไฟล์ html ขึ้นมา (5 คะแนน)

คำแนะนำ พยายามให้โค้ดของนาย B สมบูรณ์ไว้ จะได้แน่ใจว่าถูก เพราะตอนตรวจอาจารย์จะใช้ test case คนละอันกับอันที่นาย B เขียนนี้

วิธีส่ง ให้รวมไฟล์ทั้งหมดเป็นจาร์ไฟล์ไว้ใน c:\temp โดยตั้งชื่อไฟล์เป็น x_midtermProgmeth.jar โดย x คือเลข id ของ นิสิต อย่าลืมตรวจให้เรียบร้อยว่ามีซอร์สโค้ดครบถ้วน แล้วรอเจ้าหน้าที่คุมสอบมาทักอปปี เมื่อทักอปปีเสร็จแล้วจึงจะออกจากห้องได้