

Stacks and Queues

1. What is the order of growth of your RingBuffer enqueue() and dequeue()?
2. What is the order of growth of push() and pop() operations in the array implementation of Stack?
3. What is the main benefit of using a linked list implementation for stacks and queues?
4. Suppose that a client performs an intermixed sequence of (stack) push and pop operations. The push operations put the integers 0 through 9 in order on to the stack; You can pop the top of the stack off at any time. Which of the following sequence(s) of pops could occur?
 - a. 4 3 2 1 0 9 8 7 6 5
 - b. 4 6 8 7 5 3 2 9 0 1
 - c. 2 5 6 7 4 8 9 3 1 0
 - d. 4 3 2 1 0 5 6 7 8 9
 - e. 1 2 3 4 5 6 9 8 7 0
 - f. 0 4 6 5 3 8 1 7 2 9
 - g. 1 4 7 9 8 6 5 3 0 2
 - h. 0 1 2 3 4 5 6 7 8 9
5. Suppose that you use a queue to perform a sequence of enqueue and dequeue operations. The enqueue operations put the numbers 0 through 9 into the queue, in order. You can dequeue at any time. Which of the following sequence(s) of dequeues could occur?
 - a. 4 3 2 1 0 9 8 7 6 5
 - b. 4 6 8 7 5 3 2 9 0 1
 - c. 2 5 6 7 4 8 9 3 1 0
 - d. 4 3 2 1 0 5 6 7 8 9
 - e. 1 2 3 4 5 6 9 8 7 0
 - f. 0 4 6 5 3 8 1 7 2 9
 - g. 1 4 7 9 8 6 5 3 0 2
 - h. 0 1 2 3 4 5 6 7 8 9

Generics

1. Match the description to the Java code snippet

a. To-do list	i. <code>q2.enqueue(q1);</code>
b. Inserts an task into a to-do list	ii. <code>Queue<Queue<String>> q2 = new Queue<Queue<String>>();</code>
c. Retrieves a task from a to-do list	iii. <code>Queue<String> q1 = new Queue<String>();</code>
d. Can be used to reverse the characters in a word	iv. <code>q1.enqueue("Pay bills.");</code>
e. A list of to-do lists	v. <code>String s = q1.dequeue();</code>
f. Inserts a to-do list into a list	vi. <code>Stack<Character> s1 = new Stack<Character>();</code>

2. Identify which code snippets are examples auto-boxing or auto-unboxing:

<code>Integer i = 5;</code>	auto-boxing	auto-unboxing
<code>int x = i;</code>	auto-boxing	auto-unboxing
<code>Queue<Integer> q = new Queue<Integer>();</code>		
<code>q.enqueue(100);</code>	auto-boxing	auto-unboxing
<code>int x = q.dequeue();</code>	auto-boxing	auto-unboxing

3. What's wrong with each of these lines of code?

- a. `Stack<String> s1 = new Stack<String>;`
- b. `Stack<Stack<int>> s2;`
- c. `Queue<Integer[]> q;`