



## Sheet 4 Queues and Trees

**1. Suppose an initially empty queue Q has performed a total of 32 enqueue operations, 10 front operations and 15 dequeue operations, 5 of which generated EmptyQueueExceptions, which were caught and ignored. What is the current size of Q?**

**2. Consider the array-based implementation of the queue ADT.**

1. Suppose that you used the index variable r to point to the index after the last element in the queue. The front of the queue is the element at the index 0. Write algorithms to enqueue and dequeue elements from the queue. What is the problem with this implementation?
2. Alternatively, you would store the index of the first and last elements of the queue by the variables f and r. What are the values of f and r when: i. The queue is empty. ii. The queue is full. What is the problem with this scheme?
3. You then chose to solve the problem using a circular array (you would store the index of the first and last elements of the queue by the variables f and r and let f and r wrap around the array). What are the values of f and r when: i. The queue is empty. ii. The queue is full. What is the problem with this scheme?
4. You then chose to solve the problem by not allowing the last slot to be filled. How would you know that the queue is full and empty? Write the pseudocode for the Queue operations (use mod operations).

**3. Suppose you have a stack S containing n elements and a queue Q that is initially empty. Describe how you can use Q to scan S to see if it contains a certain element x, with the additional constraint that your algorithm must return the elements back to S in their original order. You may not use an array or linked list. Use only S and Q and a constant number of reference variables.**

**4. Show how to implement a stack using two queues.**

**5. Given a priority queue with the interface:**

```
void Insert(Item i, int priority);  
Item DeleteMax();
```

**Describe a how to implement a stack with the interface:**

```
void Push(Item i)  
Item Pop()
```



**6. You are given an array A of size N. You can perform an operation in which you will remove the largest and the smallest element from the array and add their difference back into the array. So, the size of the array will decrease by 1 after each operation. You are given Q tasks and in each task, you are given an integer K. For each task, you have to tell the sum of all the elements in the array after K operations.**

**Sample input:**

5 2 ← Array size is 5 and number of queries is 2  
3 2 1 5 4 ← The array elements  
1 ← First query  
2 ← Second query

**Sample output:**

13 ← Subtract  $5 - 1 = 4$ , the result array will be 3, 2, 4, 4 with sum 13  
9 ← Subtract  $5 - 1 = 4$ ,  $4 - 2 = 2$ , the result array will be 3, 2, 4 with sum 9

**What data structure would you use to solve this problem? Think of one, or invent one, then write a pseudo code for this problem.**

**7. An airport is developing a computer simulation of air-traffic control that handles events such as landings and takeoffs. Each event has a time-stamp that denotes the time when the event occurs. The simulation program needs to efficiently perform the following two fundamental operations:**

1. Insert an event with a given time-stamp (that is, add a future event)
2. Extract the event with smallest time-stamp (that is, determine the next event to process).

*Which data structure should be used for the above operations? **Why?***

**8. Write an algorithm that counts the number of leaf nodes in a binary tree. And another algorithm to count the number of internal nodes in the binary tree.**

**9. Two binary trees are equivalent if they both have the same structure with same data in corresponding nodes. Write an algorithm to test whether two binary trees T1 and T2 are Equivalent.**

**10. Write an algorithm SwapTree(T) that takes a binary tree and swaps the left and right children of every node.**