

WIA1002/WIB1002 Data Structure**Tutorial: Queue**

1. Name **four** everyday examples of a queue other than those discussed during lecture.
 - i. Cars waiting at a traffic light (First car that stops is the first one to go when the lights turn green)
 - ii. Ticket counter line (First person in line gets served first)
 - iii. Supermarket checkout counter (The customer who arrives first gets to pay and leave first)
 - iv. Patients waiting to see the doctor in clinic (First patient to arrive gets called in first)

2. What is the difference between a queue and stack?
 - A queue follows **First-In, First-out ordering** where an item is removed from the queue is the first element that was added into the queue. (First come, first served)
 - A stack follows **Last-In, First-out ordering** where an item is removed from the stack is the last element that was added into the stack.

1. Name 4 everyday examples of a queue other than those discussed during lecture.

Answer:

Waiting in line at a grocery store checkout.

Waiting for a turn at an amusement park ride.

Waiting for a table at a restaurant.

Waiting for a customer service representative on the phone

2. What is the difference between a queue and stack?

Answer:

Stacks and queues are both abstract data types that allow data to be stored and retrieved in a specific way. However, there are some key differences between the two.

A stack is a data structure that follows the Last In First Out (LIFO) principle. This means that the most recently added item is the first one to be removed. Think of a stack of plates - the last plate that was added is the first one that can be taken off the top. In a stack, two main operations are performed: push and pop. Push adds an item to the top of the stack, while pop removes the most recently added item from the top of the stack.

On the other hand, a queue is a data structure that follows the First In First Out (FIFO) principle. This means that the first item added is the first one to be removed. Think of a queue at a grocery store - the first person in line is the first one to be served. In a queue, two main operations are performed: enqueue and dequeue. Enqueue adds an item to the back of the queue, while dequeue removes the item at the front of the queue.

To summarize, the key difference between stacks and queues is the order in which items are removed. In a stack, the most recently added item is removed first, while in a queue, the first item added is removed first. It's important to understand these differences when deciding which data structure to use for a particular problem.

3. Use the following code segment to answer parts (a) through (c):

```
Queue<Integer> q = new Queue<Integer>();
Scanner keyIn = new Scanner(System.in);
for (int i = 1; i <= 5; i++)
{
    if (keyIn.nextBoolean())
        System.out.print(i + " ");
    else
        q.enqueue(i);
}
```

```
while (!q.isEmpty())
    System.out.print(q.dequeue() + " ");
System.out.println();
```

(a) What is the output for the following input sequence?

true false false true true

1 4 5 2 3

(b) Is it possible to have output: 1 3 5 4 2? If yes, give an input sequence that produces the output; or else, provide justification to your answer.

No, this is because the loop processes number sequentially from 1 to 5 which later produces two sub-lists of increasing values. If the input is true, the first increasing sequence list of numbers is printed directly (e.g. 1 3 5). Otherwise, the input is false, the numbers are enqueued and dequeued to print later in FIFO order (e.g. 2 4). Since the queue follows **First-In, First-Out (FIFO)** order, any number enqueued earlier must be printed before numbers enqueued later. Having 4 to print before 2 violates the rules of FIFO because the first number enqueued 2 must be the first to be dequeued. So, the queue would still print 2 first, forcing 1 3 5 2 4 instead.

(c) Give at least **three** input sequences that produce the output: 1 2 3 4 5

- i. true true true true true
- ii. false false false false false
- iii. true true false false false

(a) What is the output for the following input sequence?

true false false true true

Answer:

1 4 5 2 3

(b) Is it possible to have output: 1 3 5 4 2 If yes, give an input sequence that produces the output; if no, explain.

Answer:

No. The output lists in order inputs true followed by in order inputs false. The sequence 4 2 corresponds to inputs false and would have to be 2 4.

(c) Give at least three input sequences that produce the output: 1 2 3 4 5

Answer:

```
true true true true true
false false false false false
true false false false false
```

4. Hand trace a queue X through the following operations:

```
X.enqueue(new Integer(14));
X.enqueue(new Integer(3));
X.enqueue(new Integer(5));
```

```
Object Y = X.dequeue();
X.enqueue(new Integer(7));
X.enqueue(new Integer(9));
Y = X.dequeue();
X.enqueue(new Integer(2));
X.enqueue(new Integer(4));
```

Given the resulting queue X above, what would be the result of each of the following

X = {7, 9, 2, 4, 10}

Y = {5}

a) X.front();

5

b) Y = X.dequeue();

X.enqueue(new Integer(10));

X.front();

7

c) Y = X.dequeue();

Y = 7

d) X.front();

9

a) X.front(); //5 //5,7,9,2,4

b) Y = X.dequeue();

X.enqueue(new Integer(10));

X.front(); //7 //7,9,2,4,10

c) Y = X.dequeue(); //7 //9,2,4,10

d) X.front(); //9 //9,2,4,10

-
5. Provide and explain three operations / functions of Linked List based Queue that you can add to the GenericQueue class, other than those discussed in the lecture (i.e., enqueue, dequeue and getSize – from Slide 10 in the lecture slide).
- i. peek()
 - Returns the first element in the stack without actually removing it.
 - ii. isEmpty()
 - Returns true if the queue is empty or false otherwise.
 - iii. contains(E element)
 - Returns true if the given element exists in the queue or false otherwise.
5. Provide and explain three operations / functions of Linked List based Queue that you can add to the GenericQueue class, other than those discussed in the lecture (i.e., enqueue, dequeue and getSize – from Slide 10 in the lecture slide).
- isEmpty() – return true when the queue is empty
 - peek() – retrieve but do not enqueue the first element of the queue
 - toString() – convert queue into string format
 - and others (refer to the linkedlist class).