

```

1  import java.util.*;
2  // implementing the Queue ADT using dynamic array
3
4
5  class Queue{
6
7      private int array[];
8      private int maxSize; // useful to decide if resizing (doubling the array size) is
9                          // needed
10     private int endOfQueue; // same as endOfArray
11
12     public Queue(int size){
13         array = new int[size];
14         maxSize = size;
15         endOfQueue = -1;
16     }
17
18     public boolean isEmpty(){
19
20         if (endOfQueue == -1)
21             return true;
22
23         return false;
24     }
25
26     public void resize(int s){
27
28         int tempArray[] = array;
29
30         array = new int[s];
31
32         for (int index = 0; index < Math.min(s, endOfQueue+1); index++){
33             array[index] = tempArray[index];
34         }
35
36         maxSize = s;
37     }
38
39
40     public void enqueue(int data){ // same as insert 'at the end'
41
42         if (endOfQueue == maxSize-1)
43             resize(2*maxSize);
44
45         array[++endOfQueue] = data;
46     }
47
48
49
50     public int peek(){
51
52         if (endOfQueue >= 0)
53             return array[0];
54         else
55             return -1000000; // an invalid value indicating
56                             // queue is empty
57     }
58
59
60
61     public int dequeue(){
62
63         if (endOfQueue >= 0){

```

```

64         int returnVal = array[0];
65
66         for (int index = 0; index < endOfQueue; index++)
67             array[index] = array[index+1];
68
69         endOfQueue--;
70         // the endOfQueue is decreased by one
71
72         return returnVal;
73     }
74     else
75         return -1000000; // an invalid value indicating
76                         // queue is empty
77 }
78
79 }
80
81
82 class DynamicArrayQueue{
83
84     public static void main(String[] args){
85
86         Queue queue = new Queue(1);
87
88         Scanner input = new Scanner(System.in);
89
90         int queueSize;
91
92         System.out.print("Enter the number of elements you want to enqueue: ");
93         queueSize = input.nextInt();
94
95         Random randGen = new Random(System.currentTimeMillis());
96
97         int maxValue;
98
99         System.out.print("Enter the maximum value for an element: ");
100        maxValue = input.nextInt();
101
102        System.out.print("Elements enqueued: ");
103        for (int i = 0; i < queueSize; i++){
104
105            int value = randGen.nextInt(maxValue);
106            queue.enqueue(value);
107            System.out.print(value + " ");
108        }
109
110        System.out.println();
111
112        System.out.print("Elements dequeued: ");
113        while (!queue.isEmpty()){
114
115            System.out.print(queue.dequeue() + " ");
116        }
117
118        System.out.println();
119
120    }
121
122 }

```

```

Enter the number of elements you want to enqueue: 10
Enter the maximum value for an element: 50
Elements enqueued: 19 4 37 41 40 13 21 49 33 37
Elements dequeued: 19 4 37 41 40 13 21 49 33 37

```