QUESTIONS (THERE ARE TWO QUESTIONS)

Q1. *// Partial Source code for question 1*

```
public class MyLinkedList {

  public MyLinkedList() {
    head = tail = null;
  }

  private Node head, tail;

  class Node {
    String item;
    Node   next;
    Node   prev;
  }
…
…

}// End of class MyLinkedList
```

Complete the `MyLinkedList` class whose skeleton is given above by adding the methods given below. You can write other constructors, getter and setter methods, toString and equals, and other linked list related methods only if you will use them: otherwise you are not required to write them. Note: Please do not use the LinkedList class that is readily available in the Java environment.

a) public void addtoStart(String str) This method adds to the start of the list.

b) public String getElement(int i) This method returns the ith element in the list.

c) public Node removeHead() This method removes the first element and returns the reference of the Node that is deleted.

d) public Node removeLast() This method removes the last element and returns the reference of the Node that is deleted.

e) public Node remove(int i) This method removes the item at position i and returns the reference of the deleted node.

f) public void printOut() This method lists the elements in the list.

g) Write a **main** metod in which you add 5 elements using addtoStart method. Then, delete the third element and print the list. Finally, using the Stack class that is readily available in Java environment, print the list in reverse order.

Q2. You are required to add two methods to the HashTable class given in your textbook. The The first method is "**public boolean delete(String item)**" which deletes the item given from the HashTable. The other method is "**public void printHashTable()**" that prints all the elements in the HashTable. In the delete method, return false if the element is not in the HashTable, otherwise if the element is successfully deleted return true. Finally, write a Demo class in which you add some elements to HashTable, print all of the the elements, delete an element, and print all of the elements again.

Note: Hashtable.java and LinkedList2.java that it uses are attached to the project on EgeDers.

Rules for Delivering the Quizzes

1. Upload your project to the FINAL directory on EgeDers platform until **12.06.2023 Monday, 19:00.** When naming your project, use your 11 digit university student id and upload it in compressed form (such as 05110000222.rar or zip).

2. You can upload your project many times but the last uploaded version is saved.

3. The codes you write by yourself are more valuable. If codes that is prepared by plagiarizing from another student is detected, then both students will take zero (0) points and offical procedures of the University for cheating will be followed.