CE221 – Data Structures & Algorithms 03.11.2022
Fall 2022

Lab 1 – Linked Lists

Create a Java Project and implement the necessary Java classes in the same package.

1. Create a _Circular Doubly Linked List_ data structure by yourself for storing integer data. *The circularity in a linked list means that the last node (tail) of the list is connected to the first node (head) with using the links (next and prev) between them.* Your data structure must have at least two Java classes, that one of them will include the node structure of the list and the other one will include the necessary references such as the first and the last node elements of the list, as well as the size of the list (the variables are not limited to given ones, you can define more variables if your data structure needs those).
   You can design and implement the data structure any way you wanted, but remember that it must be a circular list (once again, the last node must point to the first node, or vice versa) and there must be two connections (such as the next node and the previous node) between the adjacent nodes. **[40 pts]**

2. Define a method in your linked list structure to delete both the element after the first and the element before the last nodes at the same time. Remember to change all the necessary links for the deleted nodes for arranging the new references. Make sure that your algorithm has the best running time for the worst-case scenario of the operation. **[30 pts]**

3. Define a method in your linked list structure which gets the data at a given position. Your method must have two parameters: the parameter **`int`** `position` (used for the zero-indexed position of the wanted data in the list) and the parameter **`boolean`** `isBackwards`. `isBackwards` parameter will be used to determine how the `position` is searched through the list. If the parameter is `TRUE`, then the method will count the `position` from the tail of list. If it is `FALSE`, then the method will count the `position` from the head of the list. **[30 pts]**

   EX: If your linked list is: {`head->[2-8-6-1-5]<-tail`}, using the method with `position` '3' and `isBackwards` equals to '`TRUE`', will return the number *8* to the program. If `isBackwards` was '`FALSE`' for the same position, then the returnee number would have been *1*.

   Remember that, in order to test your data structure, you need to instantiate your linked list object and populate it with real data in the main method. **Please submit all classes (including your Test class) you developed to the Blackboard system.**

Important note: Any copy from any source will result in a grade of **0**. In that case, submitting nothing at all will be more beneficial for you.

HONOR CODE:
On my honor, as an Izmir University of Economics student, I affirm that I will not give or receive any unauthorized help on this exam, and that all work will be my own. The effort in the exam belongs completely to me.