COMP 203 Data Structures and Algorithms, Fall 2024

Lab Assignment 7

Deadline: 25.11.2024 11:00 am

Read the questions and rules carefully. They are clear and well defined.

Rules:

1. **No Cheating:** You are not allowed to collaborate with your friends and use any kind of websites or AI. If your homework gives a sign of any of them, **directly it will be graded as zero**.
2. **Goal:** Please do your homework alone. Our main aim is to **learn**.
3. **Submission:** Submit your work in **one java files. DON'T USE ZIP/RAR etc. In these cases, your points will be deducted by 30%.**
4. **Coding policy:** Explain your code in comments. **This is a must!**
5. **Latency policy:** A 30% deduction will be applied for each day of late submission.

Files to submit: DLLDequeue.java

1. Implement Dequeue abstract data structure from doubly linked list. Dequeue is an abstract data structure that you can insert and remove from both front and back of the dequeue. **(100pt)**

a. Implement a Node<E> class and DLLDequeue<E> including their constructors in java. (15pt)

b. void addFirst(E element) to insert an element at the front of the dequeue. (10pt)

c. void addLast(E element) to insert an element at the back of the dequeue. (10pt)

d. E removeFirst() to remove the element at the front of the dequeue and returns it. (10pt)

e. E removelast() to remove the element at the back of the dequeue and returns it. (10pt)

f. E First() to returns the element at the front of the dequeue. (10pt)

g. E Last() to returns the element at the back of the dequeue. (5pt)

h. int size() to return number of elements in the dequeue. (5pt)

i. Boolean isEmpty() to check if the dequeue is empty or not. (5pt)

j. printDequeue() to print the elements in the dequeue. (10pt)

k. Test all your functions in the main. (10pt)

Example:

addFirst(10) //10

addFirst(20) //20,10

addFirst(30) //30,20,10

addLast(40) //30,20,10,40

printDequeue ()

30,20,10,40 //prints the result

removeFirst() //20,10,40 and returns 30

removeLast() //20,10 and returns 40

printDequeue ()

20 10 //prints the result

First() //returns 20

Last() //returns 10

size()

2 //returns the number of elements

isEmpty()

false