

# COMP 203 Data Structures and Algorithms, Fall 2024

## Lab Assignment 2

**Deadline: 14.10.2024 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 **a single java file. 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:** SinglyLinkedList.java

### 1. Singly linked List

- a. Create a generic Node<T> Class that has a constructor with parameter T data for singly linked list. (10pt)
- b. Create a generic SinglyLinkedList <T> Class that has an empty constructor. In this SLL, you have only access to head node. (10pt)
- c. Write a function with the name “public void append(T data)” to insert a node having value *data* at the end of the SLL. (15pt)
- d. Write a function with the name “public void delete(T data)” to delete the node that has the value *data*. (15pt)
- e. Write a function with the name “public void display()” to print the node values in the SLL. (15pt) It should print in the following format:  
  

```
X -> Y -> Z -> null
```
- f. Write a main function that has an Integer type SLL from SinglyLinkedList class (with name IntSLL) and do the following operations on it in the order of: (10pt)

```
IntSLL.append(1);
```

```
IntSLL.append(2);
```

```
IntSLL.append(3);
```

```
IntSLL.display();
```

```
IntSLL.delete(2);
```

```
IntSLL.display();
```

Output should be in the following form:

```
0 -> 1 -> 2 -> 3 -> null
```

```
0 -> 1 -> 3 -> null
```

g. In the same main function, create an String type SLL from SinglyLinkedList class (with name StrSLL) and do the following operations on it in the order of: (10pt)

```
StrSLL.append("Apple");
```

```
StrSLL.append("Banana");
```

```
StrSLL.append("Cherry");
```

```
StrSLL.display();
```

Display output should be in the following form:

```
Apple -> Banana -> Cherry -> null
```

h. What is the Big-O time complexity of append function where the size of SLL is n? Explain why. (5pt)

What is the Big-O time complexity of delete function where the size of SLL is n? Explain why. (5pt)

What is the Big-O time complexity of display function where the size of SLL is n? Explain why. (5pt)

Write the answers of these questions as comment lines in your code at the end of your code.