

# MTH 4300: Algorithms, Computers, and Programming II

## HW #4

Due Date: November 16th, 2024

### Problem 1

You are given two sorted linked lists list1 and list2. Merge the two lists into one sorted list. The list should be made by splicing together the nodes of the first two lists. Return the merged linked list. What is the runtime and spacetime complexity of your algorithm.**Do not use STL List**

```
LinkedList merge(LinkedList list1, LinkedList list2){...}
```

### Problem 2

Write a function to remove the nth node from the end of the list. Return true if deletion was successful, otherwise return false. What is the runtime and spacetime complexity of your algorithm.**Do not use STL List**

```
bool deleteNthNodeFromEnd(int n){...}
```

### Problem 3

Given the node:**Do not use STL List for this question**

```
struct node
{
    string first_name;
    string last_name;
    node* next;
}
```

1. Modify the link list class to sort by first name, then by last name. You must implement the sort function yourself, using the selection sort strategy. What is the runtime and spacetime complexity of your algorithm.
2. In the main function, open the file **names\_list.txt** in c++ and write your name to the file. (use fstream )
3. Read from the names\_list.txt(including your name), and create a linked list object. Each line in the file should correspond to one node, containing a first name and last name. Use the sort algorithm you wrote in part 1, to sort this linked list. Create a new file **sorted\_names.txt**, and write the sorted list to this file, one name per line.

### Problem 4

Do problem 1, 2 and 3 with STL list instead(use iterators as well).