Exercise 16 (Part B/FINAL) – The Big Three

Due Date: Today at the end of class

Turn in the whole project named **A250_E16_B_Yourlastname_Yourfirstname**

Clean up your project part A (if needed) as follows:

- **Copy constructor**
  - This is a constructor; all member variables should be initialized (3 statements).
  - Traverse the parameter list and, using the function **insertBack**, insert the data into the calling object (4 statements + 1 if you are nulling the pointer).

- **Overloaded assignment operator**
  - Implement this function in a more efficient way by considering the cases shown below.

    Given the function call

    list1 = list2;

    the overloaded assignment operator will copy the items of **list2** into an **existing** object named **list1**.

    What do you know about list1? Nothing. Therefore, you need to consider all possible cases:

    - list1 is empty
      - You will be inserting new nodes in list1 with the data copied from list2.
    - list1 has less nodes than list2
      - You will overwrite the data in all the nodes of list1 and then insert additional nodes to copy the rest of the data from list2.
    - list1 has more nodes than list2
      - You will overwrite the data in the nodes of list1 and delete the extra nodes.
    - list1 and list2 have the same number of nodes
      - Simply copy the data from one list to the other.
    - list2 is empty
      - You need to empty list1 to make it as list2, by destroying all the nodes.

    After writing the function, you will see that you might be able to combine a couple of cases, depending on how you wrote the code.

Although you have a **Main.cpp** file to check if your program executes, you do not have enough test cases to check all the necessary variables. Do **NOT** create additional test cases; **use the debugger to check** the following values for both lists:

- first
- last
- count
- address of calling object and parameter object are not the same
- address of nodes are all different