COMP 322
Winter Semester 2023

INSTRUCTOR: DR. CHAD ZAMMAR
chad.zammar@mcgill.ca

Assignment 3: Exploring Classes

Due date: 09 April 2023, 11:59 PM.

Before you start:

- Research for similar problems is tolerated. However, your submission should reflect individual work and personal effort.
- Some of the topics may not be covered in class due to our limited time. You are encouraged to find answers online. You can also reach out to the TAs for guidance.
- Please submit your assignment before the due date to avoid penalties or worse risking your assignment being rejected.
- Submit one file called **assignment3.cpp** containing all the functions together with their implementations. It will also contain the main() function that runs everything.

Make sure your code is clear and readable. **Readability of your code as well as the quality of your comments will be graded**.

- No submission by email. Submit your work to mycourse.
- If your code does not compile it will not be graded.
- Be happy when working on your assignment, because a happy software developer has more inspiration than a sad one :).

Part one (60 pts)

In this assignment we will bring the tool that you have developed in assignment 2 (git322) to the next level. We will be reviewing the whole design to make it object oriented.

You need to create a class called **Git322**. All the functions from Assignment 2 should become public methods in Git322 class.

Any global variable that you used previously has to be a private data member of class Git322.

Any helper function that you used previously has to become a private method in Git322 class.

Feel free to define any meaningful constructor for the Git322 class.

To make the design clean, the linked list that you have implemented in the previous assignment should go in its proper class, so create a class called LinkedList that contains the logic of your linked list management (adding, researching, deleting ...). Git322 class will be using the LinkedList class through a member object called **mylist**.

Please adapt the main() function and make sure that your code compiles, runs and yields the same results just like before.

If your code runs smoothly, you can now move on to the second part of this assignment.

Part two (40 pts)

Create another class called **EnhancedGit322** that inherits from class **Git322**. This class has the ability of making all the versions persistent between multiple runs of the

program. For example, if you had 3 different versions of your file and you exit the program, all the versions must be saved somewhere (either in multiple files or in one file or any other structure that you might find suitable). When you rerun your program, you should automatically have access once again to all those versions. In other words, all the versions should now be populated in memory just like before.

You may need to reimplement some (or all) of the methods inherited from Git322. Feel free to add any helper method that you find suitable.

Readapt your main() function to make use of EnhancedGit322. This is the main() function that should be active when you submit your work to mycourses.