Fall 2023

CS2001 – Data Structures

Assignment 01

Deadline: Friday, September 22 11:59 pm

Submission Information

You are required to submit your assignment on **Google Classroom**. The submission should consist of a single .zip file containing all of your code. It should be named in the following format:

DataStructures_Asg1_[Section]_[RollNo]_[Name].zip

e.g., **DataStructures_Asg1_BDS-3A_22I1234_Ali.zip**

Correct and timely submission of the assignment is the responsibility of every student. No relaxation will be given to anyone. For timely completion of the assignment, start as early as possible.

Your code must be generic and handle all errors and exceptions. If there is a syntax error in the code, zero marks will be awarded in that part of the assignment.

Plagiarism: Plagiarism is not allowed. If your assignment is found plagiarized, you will be awarded zero marks in the whole assignment category.

Learning Objective: The aim of this assignment is to have hands-on experience with linked list operations while building a practical application.

Social Media Feed

Social media feeds are at the heart of platforms like Facebook, Twitter, and Instagram, where posts are displayed in a chronological order. By using linked lists as our underlying data structure, we will explore how to manage, organize, and display posts efficiently. In this assignment, you'll learn how to create, edit, and display posts within the feed, as well as how to implement features such as searching for specific posts or sorting them by date in both ascending and descending order.

Post

Start by designing a Post class/struct that will represent individual posts. Each Post should contain the following attributes:

- ID: 4-digit unique identifier, auto-generated when a post is created
- Text: String containing the text of a post
- Timestamp: Date and time of post creation or most recent edit

Feed

Implement a Feed class/struct that will be a singly linked list of Posts. It should support the following functionality:

1. **Add a new Post**
   - Prompt the user for an input text
   - Create a new Post containing this text
   - Insert the new Post at its appropriate location in the feed
2. **Edit a Post**
   - Prompt the user for the ID of the Post to be edited along with new text
   - If ID exists in the feed, overwrite the previous text
   - Update date and time of Post according to time of editing
   - Place edited node in its correct position according to new date and time
3. **Search for Posts by text**
   - Prompt the user for input text
   - Search the Feed and show all Posts containing the input text
   - Note: You may use functions from the string library
4. **Delete a Post**
   - Prompt the user to input a Post ID
   - If ID exists in the feed, delete the Post
5. **Display all Posts (Latest first)**
   - Display all attributes of all posts in reverse chronological order
6. **Display all Posts (Oldest first)**
   - Display all attributes of all posts in chronological order

Interface

To test your application, implement an interface to perform the following operations:

- Create a Feed
- Create a Post with the following text: “Hello World!”
- Create a Post with the text: “How is everyone doing today”
- Create six Posts with random text
- Create a Post with the text: “I should be going, see you guys later.”
- Create a Post with the text: “Bye”
- Search for Post containing the word “World” and find out its ID
- Use the ID to edit this Post and change the text to “Hello guys!”
- Display all Posts in chronological order (Oldest first)
- Search all Posts containing the word “guys” to find their IDs and delete them
- Display all Posts in reverse chronological order (Newest first)

Marking Rubric

The assignment will be marked as follows:

Task		Marks
Post Structure		**25**
	Correct structure definition	10
	Auto-generation of unique ID	5
	Auto-generation of timestamp	5
	Update timestamp after edit	5
Feed Linked List		**100**
	Post insertion	5 (+5 for O(1))
	Post editing	25 (+5 for O(n))
	Post search	20
	Post deletion	10
	Display (Latest)	10
	Display (Oldest)	20 (0 if doubly linked list)
Interface		**25**
	Creating 10 posts	5
	Search and Edit	5
	Display all posts (Oldest first)	5
	Search and delete	5
	Display all posts (Newest first)	5
Total		**150**