DESIGNING AN EFFICIENT EMAIL MANAGEMENT SYSTEM

(Working Duration: 7 October 2024 – 11 November 2024 – 30 Marks)

Background:

In today's digital age, email is still one of the most essential tools for communication, both for personal and professional use. The increasing volume of email traffic has made it more difficult to manage and organize these communications effectively. An email system that is not only efficient in storing and retrieving messages, but also prioritizes important emails, handles spam effectively, and ensures timely delivery of outgoing messages is necessary for users.

The goal of this case study is to create an email system that tackles these obstacles by employing suitable data structures. By utilizing stacks and different types of queues, your team intends to develop a robust and efficient email management system capable of handling a high volume of emails while maintaining optimal performance.

Problem Context:

Several functionalities must be catered to by the proposed email system, each having its own set of requirements.

1. The **system must manage incoming emails** in a way that allows users to access their most recent messages quickly.
2. **Outgoing emails must be handled in an orderly fashion** to ensure they are sent without delay.
3. **Spam emails need to be detected and filtered effectively** to keep the inbox clean.
4. The system should be **capable of prioritizing emails based on their importance**, ensuring that critical messages receive prompt attention.

To meet these objectives, your team must design an email system in **C++,** leveraging the **stack and various types of queues** to meet the requirements of each feature.

Lab Work #2 – Program & Live Presentation Guidelines (30 Marks)

1. A team can only contain a maximum of **FOUR (4)** members.

2. Your team is required to utilize C++ programming to develop **ONE (1)** prototype in this section.

3. **Built-in containers such as <list>, <vector>, etc. are not allowed** in this assignment. All containers are self-created.

   Refer to the link: https://www.geeksforgeeks.org/containers-cpp-stl/ for further information on built-in containers in STL C++.

4. Each team member must take responsibility for **at least ONE (1)** of the following features, utilizing the *appropriate data structures and algorithms*.

   - **Inbox Management**: Store and manage incoming emails efficiently.
   - **Outbox Management**: Store and manage outgoing emails in an orderly fashion.
   - **Search and Retrieval**: Quickly search for and retrieve specific emails.
   - **Spam Detection**: Identify and filter out spam emails effectively.
   - **Priority Handling**: Manage emails based on their priority.

   Marks awarded will be based on individual contributions, considering each member's responsibility in the system and how accurately you can justify your selection of data structures and algorithms.

5. The evaluation criteria for this lab work #2 also include assessing the clarity and structural design of the code, as well as the quality of comments and adherence to good programming practices. (e.g., indentation, meaningful identifier names, comments, etc.).

6. **This task requires a group submission, but grading will be based on everyone's contribution to the system.**
   The team leader must upload a ZIP file of the system solution to the Moodle system by **11 November 2024**, Monday of Week 14, no later than 5:00 pm.

   The zip file must adhere to the following name format:

   "<GroupNo>_<student ID-leader>_<student ID-member1>_<student ID-member2>.zip"

   For example, **"G1_TP012345_TP012344_TP012123.zip"**

   Refer to **Page 6** for marking criteria of this Lab Evaluation Work #2 submission.

7. After submitting your system code to Moodle, your team must schedule a live presentation with your lecturer *between Tuesday of Week 14 and Friday of Week 15 (buffer week).*

Summary: What Do You Need to Hand in During this Assignment Submission?

1. This assignment requires **TWO (2)** submissions by your team, which include the following:

   i. Lab Work #2 – Group Submission
   - C++ solution in zip folder, inclusive *the .cpp, .hpp and csv/text files*.

   ii. Individual Live Demonstration (30 Marks)
   - Each member must present their specific contribution to the system.
   - The presentation should be completed within 30 minutes, including both the **system demonstration and a Q&A session**.
   - PowerPoint slides are not required for this demonstration.

2. Your team will need to submit all your C++ solution to the Moodle system by / on **11 November 2024,** Monday of Week 14, before / on 5.00pm and arrange schedule a live presentation with your lecturer *between 12 November 2024 (Tuesday of Week 14) and 22 November 2024 (Friday of Week 15 - buffer week).*

MARKING CRITERIA

(Lab Evaluation Work #2 - 30 MARKS)

This Lab Evaluation Work #2 will be assessed based on the following INDIVIDUAL performance criteria:

Assessment Components	**Inclusive**	**30 Marks**
CLO3: Lab Evaluation Work #2 – Individual Development Skills		
Practical Skills: Problem-Solving Skills (15 Marks)		
• Identify and address technical challenges.	Assessment of Problem-Solving Ability	
• Use of data structures and algorithms.	Technical Proficiency	
• Implementation of features according to design specifications.	Technical Proficiency	
• Code quality, including readability, efficiency, and correctness.	Code Quality Evaluation	
• Quality of individual contributions relative to team goals.	Contribution Assessment	
• Innovation and creativity in developing and implementing features.	Creativity and Innovation Evaluation	
Practical Skills: Q&A with Justification of Data Structures (15 Marks)		
• Clear and logical explanation for the choice of data structures.	Justification Ability	
• Relevance of chosen data structures to the functionality implemented.	Relevance Evaluation	
• Justification aligned with the system requirements and performance needs.	Alignment with Requirements	
• Effectiveness of the live presentation in explaining individual contributions	Presentation Effectiveness	