Dept. of Cyber Security

CS216 - Data Structures

Semester Project

(CLO 2)

Total Marks: 20

Due Date: 25th December, 2024

Instructions:

Please read the following instructions carefully before submitting the project.

- **The project will be submitted in groups of THREE students.**
- **It should be clear that your project will not get any credit if:**
- **The project is submitted after due date.**
- **The submitted project does not open or file is corrupt.**
- **The code is generated by AI!**

Submission Procedure:

Submit all project files in zip format on Google Classroom. Demos will be conducted in the Lab.

Objective:

The objective of this complex programming activity is to carry out research, investigation, analysis, design, and implementation of a real-world complex software project that has the following attributes:

1. **Depth of Analysis Required**: The activity requires abstract thinking, originality in analysis to select suitable data structures and implementing standard algorithms.

2. **Interdependence**: The activity requires designing and building high level software systems that have many interdependent components or sub-parts; file I/O, use of multiple data structures and GUI.

3. **Innovation**: The activity involves creative use of data structures principles and research-based knowledge in novel ways.

4. **Familiarity**: The activity can extend beyond previous experiences by applying programming-based approaches.

Guidelines:

- Members of each group should divide the tasks equally between them.
- While it is understood that one member cannot work on all aspects of the project, it is expected that each member should know the basic structure of the project.
- Code should be properly aligned and well commented.
- **You are NOT allowed to use STL containers or algorithms! Code everything yourself.**
- GUI should be user-friendly, practical and tidy.
- The assessment shall be done on the basis of functionality (10 marks), GUI (5 marks) and use of appropriate data structures (5 marks).

Project

Project title:

Competition Scheduler for Cyber Security Hackathons

Problem Statement (Introduction):

Hackathons and Capture-The-Flag competitions are a great way for the cyber security enthusiast to connect with fellow red-teamers and test their skills by competing against the best of the field. During the recent Pakistan Cybersecurity Challenge, the Showdown event got the attention of all, being the first ever event of its kind in the country!

Unlike general CTF events where each participant/team is competing against all other participants to achieve the maximum score, competitions like Showdown are played on one-to-one basis, just like an ICC world cup tournament.

In this project, you will build a scheduler to manage the competitions in a Showdown event.

Project requirements:

You are required to write a program which provides the functionality to meet the following requirements:

1. Each event involves 16 teams, and there are a total of 4 knock-out rounds. After the first round 8 teams go to round 2, out of which 4 teams go to round 3 (the semi-final round) and then a final is played between the last two remaining teams.
2. The scheduler reads the teams' data from a file which holds the following information about each team:
   a. Team Name
   b. Members
   c. Rank
3. Each team can have one or more members. Each team is also awarded a unique rank from 1 to 16 on the basis of a preliminary CTF round. Note that the scheduler only needs to add the team's rank; how that rank is calculated is not part of the problem.
4. The system automatically divides teams into two groups, A and B, such that eight top ranked teams are in Group A and the remaining eight are in group B. The system then schedules the first round of matches where one competitor in each match is randomly taken from group A and the other is randomly chosen from Group B. Please note that there is no use of these groups after the first round!
5. Data to be stored about a match includes names of the two teams, the date and time, status (scheduled, completed) and the winner team in case the status is *Completed*.
6. The winners of each round go to the next round and their matches are scheduled randomly.
7. The software will have a graphical user interface. Make it as simple and intuitive as you can!

8. For implementation, you'll need to select suitable data structures for tasks such as organizing and displaying scheduled matches, completed matches and their results, and for managing matches.
9. You must devise a mechanism to simulate the progress of the event so that the complete functionality of the scheduler can be demonstrated.

Stages and Timeline:

Stage 1: Design
Prepare a document which describes the overall design of the project. The report must include:

- the overall flow of the program, highlighting major components
- classes that you'll create
- data structures that you'll need
- sample screens of GUI in proper sequence

Time: 8 days – (30th November to 8th December)

Stage 2: Basic implementation
Develop the code for all the classes and implement at least half the functionality in a GUI application.
Time: 10 days – (9th December to 18th December)

Stage 3: Final submission
Develop and test the complete software with full functionality.
Time: 1 week - (19th December to 25th December)