CS261 – Data Structure and Algorithms

Term Project Guidelines

Guidelines:

1. Gitlab Repositories should be created before next lac and add nazeefulaq.uet@gmail.com as a collaborator.
2. Progress on project will be tracked from Gitlab accounts
3. Project will be evaluated based on rubrics described in the documents
4. Name of repository should be in this format: CS261F23PIDxx e.g. if project id is 9 then repository name should be CS261F23PID09

Guidelines for Project Testing:

1. Each group will prepare the following by December 22, 2023
    a. Project Configuration Document (Preferably this should be as readme file on github).
    b. You may be asked to add the students(who will perform testing of your project) as collaborators in the repository.
    c. List of assumptions assumed for algorithm and features that you have implemented.
2. Students will perform testing of assigned projects. The following issues should be detected during testing:
    a) Any type of crashes in application
    b) Issues found in the responsiveness of UI
    c) Incorrect results of algorithms
    d) Any discrepancies in the features implemented
    e) Suggestion on the improvement of UI/backend
    f) Testing methods can be automated or it can be performed manually.
    g) Try multiple inputs on algorithms to make it crash and report the scenarios which cause problem
    h) All the issues should be reported on Gitlab
3. After completion of testing, issues will be resolved and will be documented in the final report.

Criteria	Weightage	Type	2(Poor)	4(Satisfactory)	6(Average)	8(Good)	10(Excellent)
Pseudo Code of Algorithm/ Data Structures	10%	Group Evaluation	Same algorithms as in book	Made some changes in the book algorithm	Took help from some forum and written pseudo code	Algorithm designed but no design recipe followed	Totally designed your own algorithm using design approaches
Complexity of Algorithm/Algorithms	5%	Group Evaluation	Written the algorithm complexity without any calculation	Written the algorithm complexity with calculations and same as book	Written the algorithm complexity with calculations and used different method from book	Complexity calculated using the time libraries in actual code as well	Worked to reduce the code complexity and described complete process
Correctness of Algorithm/Data Structures	5%	Group Evaluation	Written the algorithm correctness same as book	Algorithm correctness with extra arguments discussions from book	Used a unique method for algorithm correctness	Used a unique method for algorithm correctness and students fully understand it.	Able to describe the algorithm correctness for any algorithms using the methods described
Team Coordination	5%	Group Evaluation	One time code commit	Code commits only for sharing the files	Regular code commits	Regular code commits with stable releases	Regular code commits with stable releases and able to resolve conflicts
Implementation of Code in High Level Language	10%	Individual Evaluation	Able to implement pseudo code without exceptions handling	Able to implement pseudo code with exceptions handling	Able to implement pseudo code with exceptions handling and proper code refactoring	Use of data storage as well with exception handing on desktop	Use of data storage, exception handling, logging and web based.
GUI for project	10%	Individual Evaluation	Incomplete GUI	Complete GUI	Responsive GUI	Responsive GUI with multiple input options	User friendly Responsive GUI
Project Report	10%	Group Evaluation	Incomplete Report	Complete Report	Complete Report without any grammatical mistake and with consistent fonts	Professional writing and document formatting	Only for top 15 professional reports.

Testing of Project	10%	Individual Evaluation	Testing without test cases	Testing with incomplete test cases	Testing with complete test cases	Testing with test cases and issues logged on github	Issues tracked on github
Creative features in Provided Project	Bonus	Individual Evaluation	Discussed new features in report that can be implemented	Worked on implementation of features with discussion	Worked on implementation of features with discussion	Worked on implementation of features with discussion	Successfully implemented new features in project
Execution	10%	Individual Evaluation	Project not running	Project running with major errors	Project running with minor errors	Project running with minor errors	Project running without errors
Viva Voce	20%	Individual Evaluation	Student able to describe project	Student able to make changes in project	Student has the some command on implemented algorithm details	Student has the complete command on implemented algorithm details	Student can answer any concept discussed in the course