TCP1101

Programming Language Fundamentals

Assignment

You may have a maximum of 4 people per group within the same lecture section.

In your code, you must use comments documenting each method (function) as to who wrote that method. (If more than one person worked on a function, you may list all their names.)

Every group must prepare a short video of less than 10 minutes presenting their system and each member briefly explains his contribution to the project. The video must include the group's structured chart and an explanation of their menu system.

Members of the group may be contacted to have a physical or video meeting with your lecturer to validate your work. You might be asked to modify the existing code to change the implementation to prove that you have wrote the sections that your name appears on.

All the programs will be checked for plagiarism, which can identify code copied from others outside your team. If you copy code from anyone else, you will be given **zero**.

You are expected to **write your own code**. If you learn and use code snippets from the internet, you must provide the reference(s).

Basic Data Analysis Application

Your system should be able to read data from a file with multiple columns separated by commas. Each file has a header that describes the data as shown and explained below:

Data.txt

5	//5 columns
Id,Age,math,science,malay	//Column names
100	//100 is the number of data rows
11920100,19,93,92,87	//data rows separated by commas
11920111,20,76,87,75	
:	
:	

Your program should be able to compute and perform the following for all the columns in the file:

1. A menu system that will allow the user to load data from a file (comma separated values - csv) and perform the required computation or generation of reports. The program must detect format errors that does not comply with the header description.
2. The menu system should respond to any wrong choice input by the user with a proper error message but should never break (crash).
3. The menu system organization and design are left for the students' creativity. For each design, the students must state the reasons behind such choice and organization.
4. The user should be able to store the results of any step (operation) in a file (text or html format).
5. The program should be able to find the following of a column or all the column. Results should be displayed as header name and values in tabular form.
    a. Minimum
    b. Maximum
    c. Median
    d. Mean
    e. Variance
    f. Standard deviation
6. Display the distinct numbers (no repetition) and a count of the occurrences of each number (frequency) in a tabular form. Your tabular form must include header titles and must be displayed and formatted nicely.
7. Plot histogram for the data in textual form.
8. Display two tables with the data items above and below the mean and displaying the number of those below and above the mean.
9. Compute the Pearson's correlation between any two selected columns showing the table of computation of the correlation (correlation measures the strength of association between two quantitative variables).
10. Compute the linear regression line formula for a user selected column or columns of the data (regression estimates the best straight line to summarize the association of data).
11. Ability to display and store data sorted (ascending or descending) using a selected column
12. A statistical report that includes all the statistical computed values above, properly formatted and stored in a text file
13. An html document of the report that can be viewed by the browser. This html report is generated by the system automatically.
14. Any function in your program must not exceed 50 lines.

Deliverables

1. C++ Source code for the entire program
    a. Every function and class must be commented to show what the purpose of that class is and the student who implemented it.
    b. All the code must be properly indented.
2. Report containing
    a. A header like this:

3. **TCP1101 Assignment**
4. Trimester 1, 2020/2021
5. *by <<TEAM NAME>>*
6. Team Leader: Name, phone number, email
7. Team members:
8. Name, phone number, email
9. Name, phone number, email
10. Name, phone number, email

    b. Instructions how to compile & run your program and user documentation on how to use your program. The lecturer marking might not have your IDE, or a different version of your IDE. **You are responsible for any loss of marks if the lecturer has trouble compiling and running your code**. (Especially be careful of capitalization of file names – Windows ignores the capitalization of file names, but LINUX and Mac do not. Some of the lecturers might be marking on LINUX or Mac.)
    c. Structed chart – showing the organization of your functions and menu design.
    d. Algorithms (pseudo code or flowchart) used in the solution with brief explanation of each one.

Zip up all the source code files together with the report (PDF) and submit to the MMLS Assignment submission system by 6 pm of the due date. **Each group submits one project, according to their MMLS group.**

Note: **All** students must be involved in the programming. You cannot say “This student just did the documentation” or “this student just did the structured chart or flowcharts”

Do not email us your project unless MMLS Assignment Submission is not working.

Late policy: 10% will be deducted if the project is submitted on 1 day late. 20% will be deducted if it is 2 days late. 30% will be deducted if it is submitted 3 days late. 40% will be deducted if it is submitted 4 days late. No submissions will be accepted after that.

If you have submitted a version, but then change it, you can re-submit until the cut-off date for each of these items, and the new version will replace the old version.

Report

Part 1:

1. Problem statement
2. Functional requirements
3. The menu system: design and motivation
4. Structured chart for the system subcomponents
5. Algorithms (flowcharts and pseudocode) and proper explanation for the computations used.

Part 2:

6. Source code with inline documentation (each function must be explained)
7. Sample input file and output generated (screen shots)
8. Video session where each student explains his contributions and a sample run of the program
9. Students must reference their sources of information about major algorithms such as correlation, regression, and other algorithms used in their programs
10. Conclusion

Due Dates

- Assignment part 1 submission date: 6th September 2020
- Complete assignment submission date: 4th October 2020
- Late policy applies until: 11th October 2020