UNIVERSITY OF SRI JAYEWARDENEPURA

Faculty of Computing

Bachelor of Computing Honours in Computer Science

First Semester Examination

October/2023

CCS1063 - Practical - Fundamentals of Computer Programming

Model Paper

Duration Four (04) Hours

INSTRUCTIONS/INFORMATION TO CANDIDATES

1. Answer **All** the questions.
2. This paper contains 1 question on 3 pages.
3. Candidates may use handwritten lecture notes (No printed materials allowed. e.g. Photocopies, Printed lecture notes, etc.)
4. Read the problem thoroughly, design and write a structured, user-friendly program in C. You are encouraged to write the comments but no marks are allocated for comments. You have to use good programming techniques that you have learned in the class.
5. Write your index number, as a comment, on the first line of your program
6. Please do not ask to explain the problem (not even a part of it) if you think something is unclear or ambiguous, make a reasonable assumption (one that does not contradict the question). Write it after your index number as a comment under the assumption you made.
7. If you have an error in your program, which you cannot correct, please ask for assistance. Some marks will be deducted.
8. Create a new program with the following information:
    a. Project title: fc110ddd(where d's are the last three digits of your index number.)
    b. Folder to create the project: Desktop
    c. File name: question01.c
9. At the end of the examination copy the folder named *fc110ddd* on the desktop to exam_nn drive. (Where nn is the machine number)

Question 1 [Total 100 Marks]: Implementing a Text File Analyzer (100 Marks)

In this practical exam, you will be tasked with creating a Text File Analyzer in C. The program should read a text file, analyze its contents, and generate statistics and insights about the text.

Part 1: Reading and Processing the File (30 Marks)

Create a C program that reads a text file provided by the user. The program should count the following statistics:

Sub-categories:

1. Total number of words in the text (10 Marks).
2. Total number of lines in the text (10 Marks).
3. The average word length in the text (10 Marks).

Part 2: Identifying Unique Words (30 Marks)

Implement a function that identifies and counts unique words in the text. A unique word is one that appears only once in the entire text.

Sub-categories:

1. Create a function to tokenize the text into words (10 Marks).
2. Implement a function to count and display unique words and their frequency (10 Marks).
3. Sort the unique words alphabetically and display them (10 Marks).

Part 3: Finding the Most Common Words (20 Marks)

Develop a function to find and display the top N most common words in the text, where N is a user-specified number.

Sub-categories:

1. Create a function to count the frequency of each word in the text (10 Marks).
2. Display the top N most common words and their frequencies (10 Marks).

Part 4: Generating a Summary Report (10 Marks)

Generate a summary report in a new text file that includes the following information:

Sub-categories:

1. Total number of words in the text (5 Mark.
2. Total number of lines in the text (5 Marks).

Part 5: Menu-Driven User Interface (10 Marks)

Build a menu-driven interface that allows the user to choose from various options like reading a file, generating statistics, finding unique words, finding common words, generating a summary report, and exiting the program.

Sub-categories:

1. Implement a menu system with appropriate options (5 Marks).
2. Ensure the program loops until the user chooses to exit (5 Marks).

Bonus (5 points):

Add additional features or functionalities to your program, such as implementing a search function to find specific words or phrases in the text handling different file encodings, or generating a word cloud based on word frequency.

****