|  |
| --- |
| **Programming Fundamentals - Lab (CL1002)** |
| **Course Instructor(s):** |
| Mr. Ammar Masood, Mam Ayesha Mariam  **Section(s): (DS-A, DS-B, AI-A, AI-B, AI-C)** |

|  |  |
| --- | --- |
| **Final Examination** | |
| **Total Time (Hrs):** | **3** |
| **Total Marks:** | **80** |
| **Total Questions:**  **Date:** Jan 1, 2025 | **3** |

**\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Roll No Course Section Student Signature**

**Do not write below this line.**

**Attempt all the questions.**

* Unzip the skeleton folder, after unzipping you will find a folder named Lab Final -> 24XXXX. Firstly, rename this folder to your roll number. For Example, 241234. Notably, there are no dashes(-) or alphabet(i) in the folder name. These are just numeric digits of your roll Number.
* Open this folder (241234 in given example) in terminal, open and work in the .cpp files it contains. At the time of submission, delete all the output files. Make sure the folder contains only the .cpp files & questions.txt file for file handling.
* Zip this folder, it will automatically become 241234.zip (as per my example, it will be different in your case). Submit this 241234.zip folder on GCR on time.
* Password for unzipping the skeleton folder is written on the last page of question paper.
* Don’t use any built in function in exam.

**[CLO 1: Demonstrate the basic concepts of programming]**

**Q1:**  **[20 marks]**

In the skeleton file q1.cpp, int main() function is given to you. You are not allowed to modify that.

**Suggestion:** You can comment the main function code temporary, and uncomment step by step as you move forward with the following tasks. **Note:** At the time of submission, the main function must be unmodified and fully uncommented.

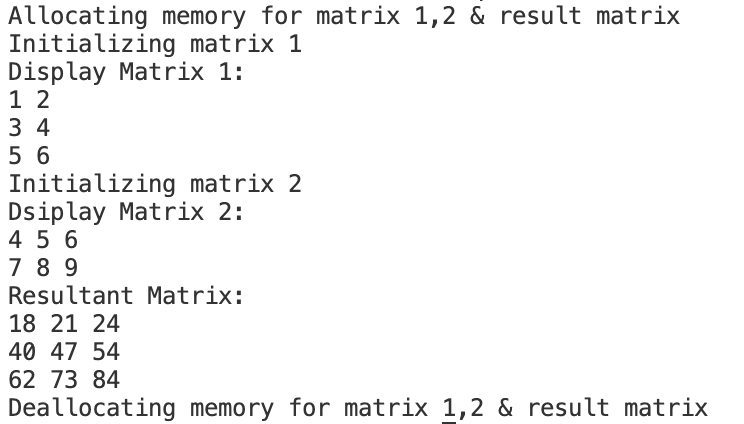
**Task1:** Make an allocateMatrix() function to allocate the 2D pointer dynamically. Function takes three arguments 1) matrix to be allocated 2) number of rows 3) Number of columns.

**Note**: InitializeMatrix() & displayMatrix() functions are already implemented. Go through with the implementation if necessary

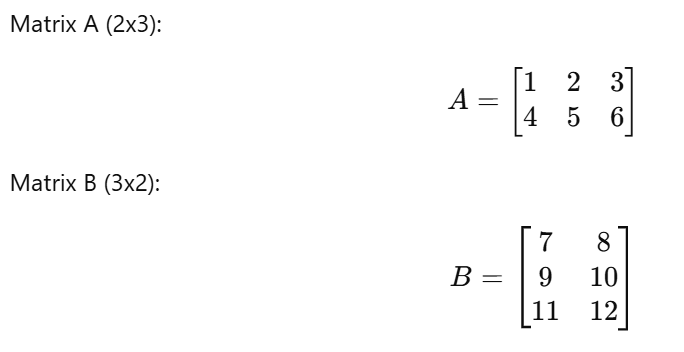
**Task2:** Make a multiplyMatrix() function to multiply two matrixes. Function takes some arguments, understand the prototype by looking at the function call in main function. **Multiply two matrixes using loops, hardcoded implementation will be a straight zero.**

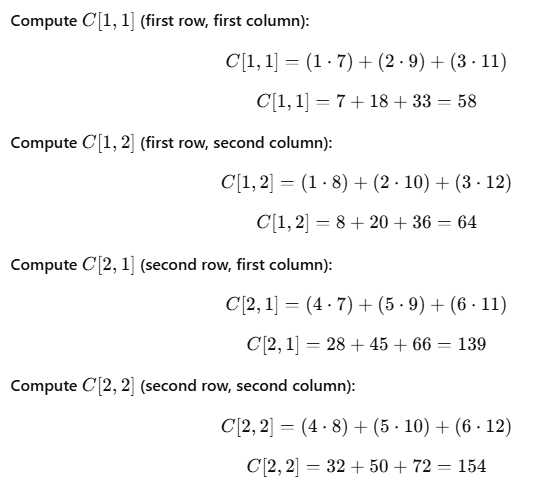
**Task3:** Implement deallocateMatrix() function to deallocate the matrixes, set the function arguments as per function call.

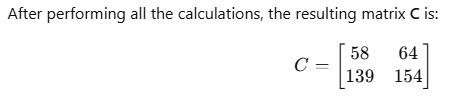
**Program Output**



**Explanation for Matrix Multiplication (It’s just an example for understanding)**







**[CLO 1: Demonstrate the basic concepts of programming]**

**Q2:**  **[20 marks]**

In this question, your task is to implement the readfile(string) function to make a console based quiz application. The int main() is already implemented, don’t modify that. In the main function, the readfile function has already been called with a filename=”questions.txt”. Your task is to implement readfile() function. The file questions.txt contains lines in the following format:

**question: option1: option2: option3: option4: correct Option**

Read the file, and prompt the user to give his answer for every question. If the answer is correct the score will be incremented by 1. If answer is incorrect the system will display **“wrong, correct option is …”**. Read all questions from the file, get user’s answer, match it with correct answer and calculate the score. At the end score must be returned.

**Note:** There are some extra strings for example: option1, option2, question etc. It is mandatory to write these strings. In short, your output must be formatted 100% like the given sample output for correctness. The text file i.e. questions.txt should remain untouched, any modification in file will result in straight zero.

Program outputs on next page. You can use the following space for rough work.

|  |  |
| --- | --- |
| **Program Output 1** | **Program Output 2** |
|  |  |

**[CLO 2: Apply algorithmic solutions related to the degree program to recent related problems]**

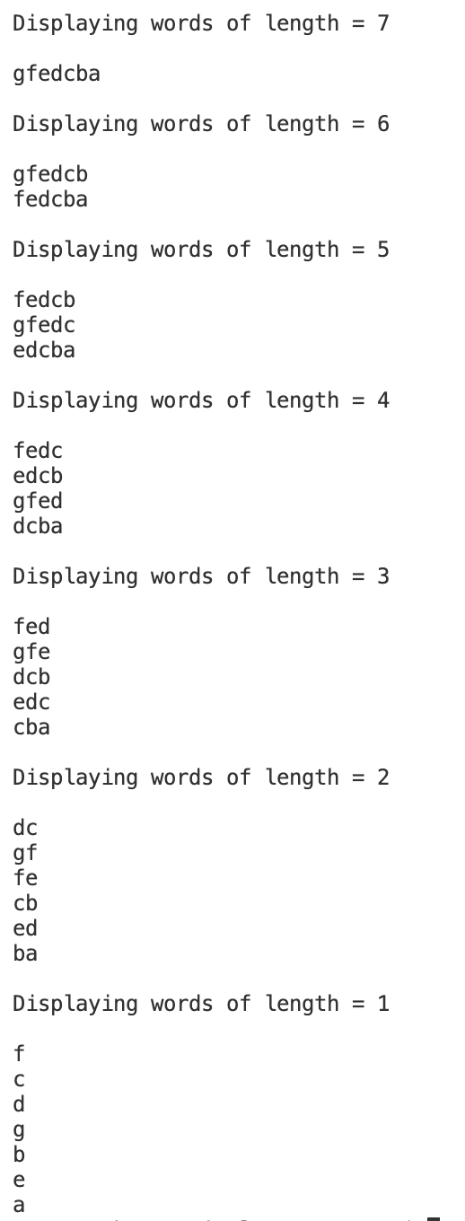
**Q3:**  **[40 marks]**

In reference to the PF Project, Word Shooter where you implemented the logic for longest words.

In this question, in the int main() the findWord() function has been called with 1D char array & array size.

1. Your task is to implement the findWord() function, in which you have to group words in **descending length (from longest to smallest).** Please note that words are made from char Array **RIGHT TO LEFT.** Moreover, there is no dictionary matching. You just have to make words. Descending order grouping is critical, in short your output should match **100%** with the sample given output.
2. Along with printing on console/terminal, your program should also create two text files i.e. **even.txt & odd.txt.** Even length Words should be **reversed & save**d in even.txt and odd length words should **reversed & saved in odd.txt Note:** Remember to delete these two TEXT files at the time of submission, when we will run this program these two files (even.txt and odd.txt) should be automatically get generated.

**Program Output (Printed on console/terminal)**



**Program Output (Text Files)**

|  |  |
| --- | --- |
| **Even.txt** | **Odd.txt** |
|  |  |

**Password:**

|  |
| --- |
| hunza871gilgit |

**Rubrics**

|  |  |  |
| --- | --- | --- |
| Question No. | Correctness | Implementation |
| 1 (total = 20) | 14 / 0 | Allocate (1.5)  Deallocate (1)  Multiply (3.5) |
| 2 (total = 20) | 15 / 0 | Reading a File (1)  Sentence Tokenization by “:” (3)  User Input & checking for correctness (1) |
| 3 (total = 40) | 31 / 0 | Making words combinations (6)  Checking for even odd (1)  Writing in file (2) |