CS 200 – Intro to Programming

Assignment 2 [Fall 2018]

Release Date: Saturday 6th October 2018, 12:00 AM
Due Date: Sunday 14th October 2018, 11:55 PM

Please keep in mind the following guidelines:

- Do not share your program code with anyone.
- Do not copy code from the internet.
- If you receive any assistance, mention the part of code in which you received assistance.
- You must be able to explain any part of your submitted code.
- All submissions are subject to automated plagiarism detection.

Submission:

You have to submit all the .cpp files containing source code. Zip all .cpp files into one file named as <your8DigitRollNumber> .zip and submit the zip file.

Part 1:

We want to implement an upper triangular matrix. It is a matrix with non-zero entries on the diagonal and non-zero entries above the main diagonal. The entries below the main diagonal are zero. Since entries below the main diagonal are zero we don't want to store them. Write a function that allocates memory to an upper triangular matrix. It will have n entries in row 1, n-1 entries in row 2, n-3 entries in row 3 and so on. (n should be input from the user).

*void **AllocateTriangularMatrix** (int ***arr,int n)*

Write a function to deallocate this matrix. Also, write a function to input the contents of this matrix from a user. Test all these functions by calling them in the proper sequence from main().

(40 Marks)

Part 2:

Read two upper triangular matrices from file. Sample is attached. File format:

Order (first line specifies the order of the matrix)
matrix 1 (values of matrix 1 in row major order)
matrix 1 (values of matrix 2 in row major order)

(10 Marks)

Part 3:

Print the upper triangular matrix row wise and column wise. It would be helpful to write a function

*int **getItem** (int ** matrix, int rowIndex, int colIndex)*

(10 Marks)

Part 4:

Find out how matrix multiplication is done. Implement this function that multiplies two upper triangular matrices A and B and stores the result in C. Make sure that for C a null pointer is passed. If it is not NULL then you should return false. Otherwise allocate the matrix C in this function. On successful multiplication the function should return true. The matrix A has dimensions m x n, Matrix B has dimensions n x r.

*bool **MultiplyMatrices** (int **A, int **B, int ***C, int m, int n, int r)*

(15 Marks)

Part 5:

Write a function to swap two rows of the upper triangular matrix. You need to think about memory allocation and deallocation carefully here.

(10 Marks)

Part 6:

Write a function to delete a row from the matrix. After this operation the matrix will no longer be a square matrix. So make it a square upper triangular matrix by deleting the last column.

(10 Marks)

Part 7:

Make sure all matrices are deallocated upon exit.

(5 Marks)

[End - Good Luck!]