CS1 Spring 2019

Assignment 2: Matrix Multiplication

Write a program that multiplies two matrices and prints the resulting matrix. Your inputs will be the two matrices. Data will be provided in a file called CS1Asg2Data.txt, and will have the form of two numbers, the number of rows and the number of columns, followed by a matrix of m rows and n columns. All numbers will be separated by spaces, so you can use “cin” for input. Once you have read all of the rows, there will be another pair of numbers m and m for the second matrix. To multiply matrices, the number of rows in one must be the same as the number of columns in the other. Once you have done the multiplication and printed the result, read the file again to look for additional matrices. If you hit the end of the file, there are no more.

Your output must show the two input matrices and the result of the multiplication, and each must be labeled. Your output should have each matrix in columns 6 characters wide, with the numbers right-justified in the field.

Since you cannot know the sizes of the arrays until you read them from the file, you must dynamically allocate them. Remember the formula for computing the offset into linear memory space from a row and column. Once you have computed and printed the result, deallocate all three matrices using the *delete*  operator.

The illustration below, from Wikipedia (https://en.wikipedia.org/wiki/Matrix\_multiplication), shows how to multiply matrices. Assume the first row of matrix A is 2 and 3 and the first column of matrix B is 3 and 4, then the (1,1) element of the result would be 2 \* 3 + 3 \* 4, or 72. That is, multiply A(1,1) by B(1,1) and add that to A(1,2) \* B(2,1).



Sample data might look like this, with the 2 2 in the first line indicating a 2x2 matrix for the first one, followed by two lines of the data for that matrix. Having read those two lines, the next 2 2 indicates that another 2x2 matrix follows:

2 2

1 2

3 4

2 2

4 5

6 7

When you print the input matrices, your output should look like this, followed by the product.

Matrix 1

1. 2
2. 4

Matrix 2

1. 5

6 7

The grader may test your program with data other than what is in the test file, so your program must be general enough to work with any good data. However, the program will not be tested with data that is not numbers, or not in the format given. It is possible that your program will be tested with data where the number of rows in one matrix is not equal to the number of columns in the other, in which case you should print an error message. Once you have computed the product for the first two matrices, read the file for data for another set. See the sample data for this.

Name your .CPP file <NetID>Asg2.cpp

**Hand in your .CPP file through eLearning.**

|  |  |
| --- | --- |
| **Grading criteria:** | |
| Program runs correctly for good data and rejects bad data, as explained above | 40 |
| Program is structured well, using functions as needed rather than in-line code | 35 |
| Program uses dynamic arrays for the squares | 15 |
| Program comments and variable names. Make sure you have both header comments and internal comments as needed. | 10 |