CS211 Homework – 4

Total Marks: 100

Due Date: 10/16/2019

This homework assignment has **2 parts** which require submitting **2 separate programs** to blackboard.

Question 1 of 2. **Count Frequency**: *[50 marks]* Write a function that counts the frequencies of array elements.

Your program will take an array as input from the user. Write a function called CountFreq(…) that takes the array as a parameter and print all elements and their frequencies. You can print the elements in any order.

Make sure your code works for any input number, not just the test cases. Your code will be tested on other test cases not listed here.

Please properly comment your code before submission.

For this part of the assignment, name your source file as CountFrequency_WSUID.cpp. For example, if your user ID is A999B999 name your file as CountFrequency_A999B999.cpp.

Sample Test Cases:

Test Case 1:	**Test Case 2:**
Input: Size of array: 4 Enter Element 0: 1 Enter Element 1: -7 Enter Element 2: -7 Enter Element 3: 2 **Output:** number -> count 1 -> 1 -7 -> 2 2 -> 1	**Input:** Size of array: 10 Enter Element 0: 2 Enter Element 1: 3 Enter Element 2: 2 Enter Element 3: 2 Enter Element 4: 2 Enter Element 5: 3 Enter Element 6: 3 Enter Element 7: 2 Enter Element 8: 2 Enter Element 9: 2 **Output:** number -> count 2 -> 7 3 -> 3

CS211 Homework – 4

Total Marks: 100

Due Date: 10/16/2019

Question 2 of 2. **Matrix Multiplication**: *[50 marks]* Write a function which can multiply matrices and returns the result matrix. A matrix is defined as a 2-dimensional array.

Given two 2-dimensional arrays **A** and **B**. if **A** is an *n* × *m* matrix with *n* rows and *m* columns and **B** is an *m* × *p* matrix, their matrix product **AB** is an *n* × *p* matrix, in which the *m* entries across a row of **A** are multiplied with the *m* entries down a column of **B** and summed to produce an entry of **AB**. The number of columns of **A** should be the same as the number of rows in **B**.

$$A \times B = \underset{\underline{2 \times 3\ matrix}}{\begin{array}{|c|c|c|}\hline A_{0,0} & A_{0,1} & A_{0,2} \\ \hline A_{1,0} & A_{1,1} & A_{1,2} \\ \hline \end{array}} \times \underset{\underline{3 \times 3\ matrix}}{\begin{array}{|c|c|c|}\hline B_{0,0} & B_{0,1} & B_{0,2} \\ \hline B_{1,0} & B_{1,1} & B_{1,2} \\ \hline B_{2,0} & B_{2,1} & B_{2,2} \\ \hline \end{array}} =$$

$$\underset{\underline{2 \times 3\ matrix}}{\begin{array}{|l|l|l|}\hline (A_{0,0} * B_{0,0}) + (A_{0,1} * B_{1,0}) + (A_{0,2} * B_{2,0}) & (A_{0,0} * B_{0,1}) + (A_{0,1} * B_{1,1}) + (A_{0,2} * B_{2,1}) & (A_{0,0} * B_{0,2}) + (A_{0,1} * B_{1,2}) + (A_{0,2} * B_{2,2}) \\ \hline (A_{1,0} * B_{0,0}) + (A_{1,1} * B_{1,0}) + (A_{1,2} * B_{2,0}) & (A_{1,0} * B_{0,1}) + (A_{1,1} * B_{1,1}) + (A_{1,2} * B_{2,1}) & (A_{1,0} * B_{0,2}) + (A_{1,1} * B_{1,2}) + (A_{1,2} * B_{2,2}) \\ \hline \end{array}}$$

Your code should ask the user for the matrix size and then take each matrix as input. Your program will check if the two matrices can be multiplied. Following this, if the matrices are compatible for multiplication, your program will print the resultant array from the multiplication. Your program will contain a user defined function MatrixMultiply(…). Your program shall pass the matrices as parameters to the function MatrixMultiply(…), which will multiply the matrices and return the resultant matrix as a reference.

Make sure your code works for any input number, not just the test cases. Your code will be tested on other test cases not listed here.

Please properly comment your code before submission.

For this part of the assignment, name your source file as MatrixMultiply_WSUID.cpp. For example, if your user ID is A999B999 name your file as MatrixMultiply_A999B999.cpp.

Sample Test Cases:

Test Case 1:

Input:

```
Input Matrix 1:
Number of rows: 1
Number of columns: 2
Enter Element [0, 0]: 10
Enter Element [0, 1]: 20
Input Matrix 2:
Number of rows: 2
Number of columns: 1
Enter Element [0, 0]: 30
Enter Element [1, 0]: 40
```

Output:

```
Resultant Matrix: 1 X 1
1100
```

Test Case 2:

Input:

```
Input Matrix 1:
Number of rows: 2
Number of columns: 3
Enter Element [0, 0]: 2
Enter Element [0, 1]: 4
Enter Element [0, 2]: 6
Enter Element [1, 0]: 5
Enter Element [1, 1]: 2
Enter Element [1, 2]: 1
Input Matrix 2:
Number of rows: 3
Number of columns: 4
Enter Element [0, 0]: 4
Enter Element [0, 1]: 2
Enter Element [0, 2]: 2
Enter Element [0, 3]: 1
Enter Element [1, 0]: 1
Enter Element [1, 1]: 2
Enter Element [1, 2]: 3
Enter Element [1, 3]: 2
Enter Element [2, 0]: 5
Enter Element [2, 1]: 2
Enter Element [2, 2]: 1
Enter Element [2, 3]: 2
```

Output:

```
Resultant Matrix: 2 X 4
42 24 22 22
27 16 17 11
```

Test Case 3:

Input:

```
Input Matrix 1:
Number of rows: 1
Number of columns: 2
Enter Element [0, 0]: 10
Enter Element [0, 1]: 20
Input Matrix 2:
Number of rows: 1
Number of columns: 2
Enter Element [0, 0]: 30
Enter Element [0, 1]: 40
```

Output:

```
Resultant Matrix: Matrices not compatible
```

Test Case 4:

Input:

```
Input Matrix 1:
Number of rows: 1
Number of columns: 1
Enter Element [0, 0]: 10
Input Matrix 2:
Number of rows: 1
Number of columns: 1
Enter Element [0, 0]: 30
```

Output:

```
Resultant Matrix: 1 X 1
300
```