

It is required to design and implement a matrix calculator that supports all necessary operations that can be done on Matrices. Operations that will be supported in this first milestone are: 1- Addition of two matrices 2- Subtraction of two matrices 3- Multiplication of two matrices 4- Multiplying matrix by a scalar

REMOVED THE DIVISION

Input Format

The matrix should be entered by the user in the following format: [1 2 5, -1 8 14.2, 7.01 -18 99.3] which is the format equivalent to a 3x3 matrix as shown:

$$\begin{bmatrix} 1 & 2 & 5 \\ -1 & 8 & 14.2 \\ 7.01 & -18 & 99.3 \end{bmatrix}$$

User can choose between the different operations as follows:

- In case of Addition: user uses '+' symbol between the two matrices as follows:

```
[1 2 4, 4 7 8]
+
[-5 3 5, 7 6 1]
```

The first matrix is in a line, then the operator on a new line, and finally the second matrix is also on a new line.

The output must be in the following format: (Will be printed on a whole new line)

```
[1 2 4, 4 7 8]
+
[-5 3 5, 7 6 1]
[-4 5 9, 11 13 9]
```

- In case of subtraction: user uses '-' symbol between the two matrices.
- In case of multiplication: user uses '*' symbol between the two matrices:

1. In case of multiplication of a matrix by a scalar, the user can begin with either the matrix or the scalar (as he wishes), with the multiplication operator between them.

```
[1 2 4, 4 7 8]
*
-1
[-1 -2 -4, -4 -7 -8]

4
*
[1 2 4, 4 7 8]
[4 8 16, 16 28 32]
```

2. In case of multiplication of a matrix by another matrix, the input is done as similar to the addition and subtraction step. (But be aware of the dimensions in multiplication!!)

REMOVED DIVISION

Constraints

User will input string as shown previously:

Example for a 3x4 matrix, it should be written as follows [-0.1 2 5, -1 8 14.2, 7.01 -18 99.3, 15.0 -99.24 18]

User will input the matrix of size nxm; where

$0 < n < 100$ and $0 < m < 100$

The matrix will have numbers constrained by the following: each value v inside the matrix lies in the range of

$-9999.999 < v < 9999.999$

Important Note

- If the user enters a matrix in an incorrect format (other than the stated input format) your program must detect that and output the word "ERROR!" (MUST BE WRITTEN AS SHOWN, don't forget the exclamation mark, and note that all letters are capitals) For Example:

```
[1 2 4, 4 7, 8]
+
[-5 3 5, 7 6 1]
ERROR!
```

- Your program should validate the mathematical conditions to do the different operations, for example: In addition and subtraction, both matrices should have the same dimensions. If not, your program should output an "ERROR!" message. Also, in case of multiplication for example, your program should check that the dimensions are compatible with each other and that the multiplication can take place correctly. If not, your program should output an "ERROR!" message. For Example:

```
[1 2 4, 9 3 7]
+
[1 9, 2 3, 4 7]
ERROR!
```

Output Format

The output should be a matrix of the result printed on a new line as shown previously.

Also, the error message should be in the following format: "ERROR!" on a new line as shown in the examples above.

NOTE: In the case of decimal point outputm you should set the precision of the output to 2 decimal point places

Sample Input 0

```
[1 2 3, 4 5 6, 7 8 9]
+
[0 4 7, 3 4 7, -1 15 20]
```

Sample Output 0

```
[1 6 10, 7 9 13, 6 23 29]
```

Sample Input 1

```
[5 1 2]
-
[5 1 2]
```

Sample Output 1

[0 0 0]