

Map Reduce Advanced - Matrix Multiplication

Mappers and Reducers

[Here's](#) a quick but comprehensive introduction to the idea of splitting tasks into a MapReduce model. The four important functions involved are:

```
Map (the mapper function)
EmitIntermediate(the intermediate key,value pairs emitted by the mapper functions)
Reduce (the reducer function)
Emit (the final output, after summarization from the Reduce functions)
```

We provide you with a single system, single thread version of a basic MapReduce implementation.

Task

The input is a number of test cases with two matrices each. A single test case will look like:

```
[#Rows_Matrix_1] [#Columns_Matrix_1]
[Row_1_Matrix_1]
.
.
[Row_N_Matrix_1]
[#Rows_Matrix_2] [#Columns_Matrix_2]
[Row_1_Matrix_2]
.
.
[Row_N_Matrix_2]
```

The required output is to print the product of the two matrices in the format shown. The code for the MapReduce class, parts related to IO etc. has already been provided. However, the mapper and reducer functions are incomplete. Your task is to fill up the mapper and reducer functions appropriately, such that the program works, and outputs the product of the two matrices, in row-wise manner.

Also, this program outputs certain information to the error stream. This information has been logged to help beginners gain a better understanding of the the intermediate steps in a map-reduce process.

Languages Supported

Currently, we provide the base code in Python.

Input Format

First line of the input will contain the number of test cases, for each test case, there will be two matrices. For each matrix, the first line will contain the number of rows and columns and from the second line, row*column number of elements of matrix will be given. We have already written the input handling code to read in this data.

Output Format

Again, the output handling part has already been provided in the template code. The output contains the product matrix arranged in a row-wise manner.

Sample Input

```
1
3 2
1 2
2 3
4 5
2 3
2 4 5
3 6 7
```

Sample Output

```
8 16 19
13 26 31
23 46 55
```

Explanation

The product of the two matrices given in the input sample is the matrix given in output sample.