

## Assessment Tool

Create two programs (`lab08_<surname>_A.c` and `lab08_<surname>_B.c`) that will communicate using a shared memory to compute for the matrix product of two matrices.

`lab08_<surname>_A.c` should read a file named `input.txt` (format below) for the number of test cases and the matrices to be multiplied. `lab08_<surname>_A.c` will compute the left half of the answer while `lab08_<surname>_B.c` will compute the right half of the answer. If the number of columns for the answer is odd, `lab08_<surname>_A.c` will compute for the values in the excess column. Once every cell in the answer is computed, `lab08_<surname>_B.c` must print in the terminal the matrix product.

**Note:** Your program **MUST** check if matrix multiplication can be done on the given matrices. That is, if the number of columns of Matrix A is not equal to the number of rows of Matrix B, multiplication can't be done on the two matrices, and your program must proceed to the next test case, if any.

**FORMAT OF** `input.txt`:

```
N                // # of test cases
rowA colA        // # of rows and cols of Matrix A
e00 e01 e02 ...  // elements of Matrix A
e10 e11 e12 ...
...
rowB colB        // # of rows and cols of Matrix B
e00 e01 e02 ...  // elements of Matrix B
e10 e11 e12 ...
...
...              // other test cases, if any
```

**SAMPLE** `input.txt` file:

```
2
5 5
1 2 3 4 5
2 3 4 5 6
3 4 5 6 7
4 5 6 7 8
5 6 7 8 9
5 3
1 2 3
2 3 4
3 4 5
4 5 6
5 6 7
5 5
1 2 3 4 5
2 3 4 5 6
3 4 5 6 7
4 5 6 7 8
5 6 7 8 9
5 4
1 2 3 4
2 3 4 5
3 4 5 6
4 5 6 7
5 6 7 8
```

**SAMPLE** output on the terminal:

```
55  70  85
70  90 110
85 110 135
100 130 160
115 150 185

55  70  85 100
70  90 110 130
85 110 135 160
100 130 160 190
115 150 185 220
```