

# CS314 HW 1: Imperative programming and Python

Spring 2019

Implement the following in Python. You should not use any libraries that do all the heavy lifting for you.

1. **matmult.py:** Matrix multiplication. Two matrices will be provided via stdin, first with the number of rows and columns, followed by the matrix values, separated by a single space:

Example input (a  $2 \times 3$  matrix and a  $3 \times 1$  matrix):

```
2 3
1.2 2.3 3.4
4.5 5.6 6.7
3 1
6.54
5.43
4.32
```

If the two matrices can be multiplied, print the resulting matrix in the same format. If they cannot be multiplied, print “invalid input”. You can assume the input will always be formatted correctly, and no matrix will have 0 rows or 0 columns.

2. **bst.py:** Implement a binary search tree class (without balancing). Commands will be given one per line via stdin – **i** for insert and **q** for query.

For insert commands, you should not print any output.

For query commands, you should report “not found” if the value was not found. If it was found, you should report “found: root” if it’s the root element, or the sequence of left/right edges to follow to reach it otherwise, e.g., “found: r r l r”.

Example:

```
i 1
i 2
```