

Implement Dijkstra's algorithm to find the shortest path to every other vertex. The input text file will contain the graph represented as an adjacency matrix. Values of 2 million will represent "infinity" edge weight (when there is no edge between the vertices). You can assume all shortest paths will be less than 2 million.

The first line in the input file is the source vertex, one space, then the destination vertex (with the first row of the adjacency matrix being vertex "0", the second row being vertex "1", and so on). Afterwards is the adjacency matrix (which will be square). Your output.txt should contain the shortest path length from this source to destination vertex followed by a ":". After the ":" should be the actual list of vertexes you should travel through for this shortest path.

Sample input file (first line always source and destination vertex, afterwards is adjacency matrix):

```
0 1
0 2000000 4
2 0 7
2000000 3 0
```

Sample output.txt (7 is the shortest path length (before ":"), 0 2 1 is the actual path (after ":")):

```
7: 0 2 1
```