

UCF Local Contest — September 5, 2015

Longest Path

filename: longpath
(*Difficulty Level:* Hard)

You are visiting a lovely garden, with several points of interest. Every pair of points of interest have a single, one way path (edge) connecting them. (The paths are one way so that guests don't get upset having to deal with other guests walking in the opposite direction.) Luckily, you get to decide which point of interest to begin from. Naturally, to get your money's worth, you'd like to visit as many of the points of interest as possible without revisiting any point of interest, since that would be a waste of time. Determine a walking path that allows you to visit the maximal number of points of interest without repeating any of them.

The Problem:

Given a directed graph containing exactly one edge (path) between every pair of vertices (points of interest), determine a walking path of maximal length that doesn't revisit any vertex.

The Input:

The first input line contains a positive integer, n ($n \leq 100$), indicating the number of gardens to check. The input for each garden follows. Each set starts with a line with a single integer, v ($1 \leq v \leq 500$), the number of points of interest for that case. The following v lines will contain v space separated integers each. The j^{th} ($1 \leq j \leq v$) of these integers on the i^{th} ($1 \leq i \leq v$) line is 1 if there is a path from point i to point j , or 0 if there is a directed path from point j to point i instead. The i^{th} integer on the i^{th} line will always be 0. It is guaranteed that the input information is consistent - if the j^{th} integer on the i^{th} line is 1 then the i^{th} integer on the j^{th} line is 0, and vice versa.

The Output:

For each garden, on a line by itself, output a space separated list of any maximal length sequence of unique points of interest to visit.

Sample Input:

```
2
4
0 1 1 0
0 0 0 1
0 1 0 0
1 0 1 0
3
0 1 1
0 0 1
0 0 0
```

Sample Output:

```
1 2 4 3
1 2 3
```