Instructions

The point of this assignment is to help us understand how you think, code, test, and communicate. You may craft your solution in either Java, Objective C, or Swift. Your deliverable is the source code and project files to your solution along with instructions on how we run your code. It is important that your code is test driven using a unit/integration test framework.

All problems require some kind of input. You are free to implement any mechanism for feeding input into your solution (for example, using hard coded data within a unit test). You should provide sufficient evidence that your solution is complete by, as a minimum, indicating that it works correctly against the supplied test data.

Although there is a right answer for the data, there is no right answer about presentation. Your submission should be written in a way that you find satisfying for yourself. When crafting your solution, remember that we value TDD-ed code, and spend more time reviewing your code than checking out its GUI goodness.

Path of Lowest Cost

The Challenge

The objective is to find the path of lowest cost when moving across a grid. For this challenge, you are provided a grid of integers where each integer represents the amount of cost encountered at a given point on the grid. A path enters the grid from the left (at any point) and passes through the grid to the right, moving only one column per round. Movement is always to the same row or an adjacent row, meaning the path can proceed horizontally or diagonally. For the sake of this challenge, we assume the first and last row are also adjacent. Effectively, the grid "wraps".
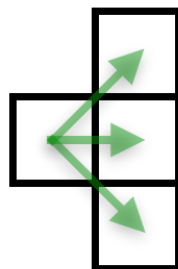
The total cost of a path is the sum of the integers in each of the visited cells. The solution needs to handle grids of various sizes with a minimum of 1 row and 5 columns up to 10 rows and 100 columns. If in the next move, the total cost will exceed 50, that path is abandoned.

The purpose of this challenge is to find the path of least cost (that is, the path with the lowest total cost of any possible path). The paths of least cost through two slightly different 5 x 6 grids are shown below. The grid values differ only in the bottom row. The path for the grid on the right takes advantage of the adjacency between the first and last rows.
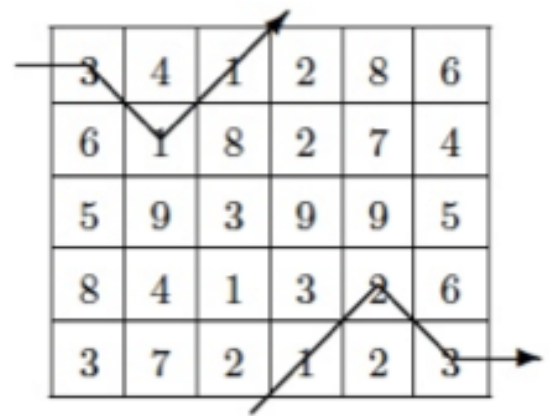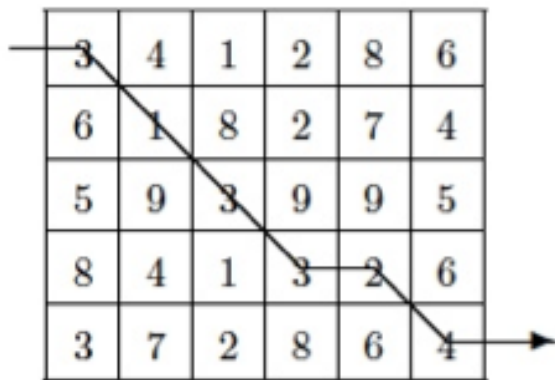
Input

The input consists of a sequence of row specifications. Each row is represented by a series of delimited integers on a single line. Note: integers are not restricted to being positive.

Output

Three lines should be output for each matrix specification. The first line is either "Yes" or "No" to indicate the path made it all the way through the grid. The second line is the total cost. The third line shows the path taken as a sequence of n delimited integers, each representing the rows traversed in turn. If there is more than one path of least cost, only one path need be shown in the solution.

Example	Input	Output
One	3 4 1 2 8 6	Yes
	6 1 8 2 7 4	16
	5 9 3 9 9 5	1 2 3 4 4 5
	8 4 1 3 2 6	
	3 7 2 8 6 4	
Two	3 4 1 2 8 6	Yes
	6 1 8 2 7 4	11
	5 9 3 9 9 5	1 2 1 5 4 5
	8 4 1 3 2 6	
	3 7 2 1 2 3	
Three	19 10 19 10 19	No
	21 23 20 19 12	48
	20 12 20 11 10	1 1 1