#Problem 1: LIS, Longest Increasing Subsequence

Description:
You will be given an array of size N having N integer numbers. You need to calculate the longest increasing subsequence length of the array.

In the first line you will be given N. In the following line you will have N integer numbers.

Limits
1<=N<=30

Test Cases:

Input	Output
3 1 2 3	3
5 1 100 2 3 4	4
7 1 100 2 105 7 109 111	5
10 9 2 5 3 7 11 8 10 13 6	6
15 46 33 66 34 65 4 28 32 41 70 51 2 35 11 26	5

#Problem 2: Hill Climbing

Description:
An avenger is trying to climb a very dangerous hill. There are some places on the hill which are very steep and dangerous to climb. Also, there are some places on the hill which are pretty flat and possess no difficulties.

This hill climbing problem can be modeled as a 2D grid problem. Where the grid has M rows and N columns. Analogically here, M denotes the height and N denotes the width of the hill respectively.

The climber is at the bottom now. He will start climbing the hill aka start traversing the grid from the bottom row. In the grid, for each cell a value is given which denotes the danger lying in that cell. If the climber comes in this cell, it will add danger for him in the path to climb the hill. In this problem, you need to find the optimal path for the climber to reach the top of the hill,

minimizing the total danger of climbing. From each cell, a climber can make three possible moves which has been shown in the following figure,

**climbed complete hill*

Can go here	Can go here	Can go here	
	Current cell		

**hill bottom*

In the first line, you will be given two values M and N denoting the number of rows and columns of the grid. Then you will have M lines of values where each line will have N values. Here the values lying in the i^{th} line denote the values of the i^{th} row of the grid. In such i^{th} line, j^{th} value denotes the danger value of grid cell (i,j).

In the output, you need to print the minimum possible danger value for the climber to reach the top row. Outside of the bottom row means, he did not start climbing and outside of the top row means, he has finished climbing. The climber can not move right from any cells belonging to the rightmost column, similarly can not move left from the leftmost column, because it would bring death to him.

Limits
1<=M,N<=40
0<=Grid cell [i,j] <=1000 for all (i,j) pairs

Test Cases:

Input	Output
3 3 2 10 0 5 0 7 0 6 4	0
4 3 0 2 1 9 8 2 3 2 7 10 5 2	7
3 3 14 25 0	6

1 5 10 5 1 2	
3 4 1 4 4 4 4 4 1 4 2 1 4 3	6
10 10 12 96 71 28 53 50 24 83 99 30 73 70 37 97 99 1 32 99 84 13 73 13 15 49 69 50 96 3 55 17 32 77 47 41 51 75 3 80 44 87 16 20 2 64 20 53 97 54 17 93 14 7 99 72 66 4 4 84 4 28 47 57 56 12 86 65 10 19 59 99 51 61 31 96 7 5 31 37 86 6 78 37 55 86 99 11 72 13 17 51 87 86 32 11 52 1 78 1 34 76	146

#Problem 3: MCM Calculation

Description:
You will be given N number of 2D matrices' information aka row and column value for each matrix. You need to calculate the minimum number of operations needed to calculate the multiplication of all the matrices.

In the first line you will be given N. In the following N lines you will find the row and column information of each given matrix, ith line contains the information of ith matrix.

Test Cases:

Input	Output
2 1 2 2 3	6
3 1 2 2 3 3 5	21
3 10 30 30 5	3500

5 40	
4 1 5 5 7 7 10 10 5	155
5 1 47 47 5 5 30 30 40 40 23	2505
6 1 7 7 35 35 42 42 3 3 2 2 65	1977