y2018-1-3. Динамическое программирование

A. Grasshopper and Money

2 seconds, 256 megabytes

The grasshopper is traveling from cell 1 to cell n. At the beginning, the grasshopper sits on cell 1. The cells are numbered from 1 to n. He can move from 1 to k cells forward in one jump.

In each cell grasshopper can get or lose several gold coins (for each cell this number is known). Determine how the grasshopper needs to jump to maximize the total number of coins in the end. Consider that the grasshopper cannot jump backwards.

Input
The first line contains two integers n and k ($3 \le n \le 10\,000$, $1 \le k \le 10\,000$) — the number of cells and the largest jump. The second line contains $n - 2$ integers, the number of coins that the grasshopper gets on each cell, from the second to the $n - 1$-th. If this number is negative, the grasshopper loses coins. All the numbers do not exceed $10\,000$ by absolute value.

Output
In the first line output the maximal number of coins. In the second line output the number of jumps. In the third line output the cells visited by the grasshoper.

input
```
5 3
2 -3 5
```
output
```
7
3
1 2 4 5
```

input
```
10 3
-13 -2 -14 -124 -9 -6 -5 -7
```
output
```
-16
4
1 3 6 8 10
```

input
```
12 5
-5 -4 -3 -2 -1 1 2 3 4 5
```
output
```
14
7
1 6 7 8 9 10 11 12
```

B. Turtle and Money

2 seconds, 256 megabytes

There is a rectangular field of size $n \times m$. The turtle wants to move from cell $(1, 1)$ to cell (n, m), in one step she can move to the next cell down or right. For each passed cell, the turtle gains (or loses) several gold coins (this number is known for each cell).

Determine what the maximum number of coins Turtle can collect on the way and how she needs to go for it.

Input
The first line contains two integers: n and m ($2 \le n, m \le 1000$). Each of the following n lines contains m numbers a_{ij} ($|a_{ij}| \le 10$), which indicate the number of coins received by the turtle on each cell. If this number is negative, the turtle loses coins.

Output
In the first line, output maximal number of coins that turtle can collect. In the second output the commands to be executed by the turtle, without spaces : the letter «R» indicates a step to the right, and the letter «D» a step down.

input
```
3 3
0 2 -3
2 -5 7
1 2 0
```
output
```
6
RRDD
```

input
```
4 5
4 5 3 2 9
4 6 7 5 9
5 2 5 -3 -10
3 5 2 9 3
```
output
```
41
RDRDDRR
```

Statement is not available on English language

C. Наибольшая возрастающая подпоследовательность

2 секунды, 256 мегабайт

Пусть $a_1, a_2, \ldots, a_n$ — числовая последовательность. Длина последовательности — это количество элементов этой последовательности. Последовательность $a_{i_1}, a_{i_2}, \ldots, a_{i_k}$ называется подпоследовательностью последовательности a, если $1 \le i_1 < i_2 < \ldots < i_k \le n$. Последовательность a называется возрастающей, если $a_1 < a_2 < \ldots < a_n$.

Вам дана последовательность, содержащая n целых чисел. Найдите ее самую длинную возрастающую подпоследовательность.

Входные данные
В первой строке задано одно число n ($1 \le n \le 2000$) — длина подпоследовательности. В следующей строке задано n целых чисел a_i ($-10^9 \le a_i \le 10^9$) — элементы последовательности.

Выходные данные
В первой строке выведите число k — длину наибольшей возрастающей подпоследовательности. В следующей строке выведите k чисел — саму подпоследовательность.

входные данные
```
8
1 4 1 5 3 3 4 2
```
выходные данные
```
3
1 4 5
```

входные данные
```
3
1 2 3
```

выходные данные

```
3
1 2 3
```

D. Knight and Phone

1 second, 256 megabytes

The chess association decides to distribute to its members special phone numbers that can be obtained by the moves of a knight on the following numpad:

1	2	3
4	5	6
7	8	9
	0	

For example, 340-49-27 is one of these numbers. Please note that a phone number cannot begin with 0 and 8.

Please calculate the number of such phone numbers of length n. Since this number can be very large output it modulo 10^9.

Input

The sole line of the input contains one integer n $(1 \le n \le 100)$ — the length of the phone numbers.

Output

The sole line of the output should contain one integer — the answer to the problem.

input

```
1
```

output

```
8
```

input

```
2
```

output

```
16
```

E. Levenshtein Distance

2 seconds, 256 megabytes

You are given a string. You can perform operations of three types:

1. Replace one symbol by another.
2. Delete any symbol.
3. Insert any symbol in any position of the string.

The smallest number of operations to be performed in order to transform one string to another one is named edit distance or Levenshtein distance.

Find the Levenshtein distance between two given strings.

Input

The input contains two lines that contain two strings. The length of each string does not exceed $1\ 000$ symbols and they consist of only uppercase Latin letters.

Output

The sole line of the output should contain the Levenshtein distance between two given strings.

input

```
ABCDEFGH
ACDEXGIH
```

output

```
3
```

F. Cafe

2 seconds, 64 megabytes

A new cafe was opened near the University. It has the following discount scheme: if you buy on more than 100 roubles you get a flyer for a free dinner.

You want to buy dinner each day out of n and you know the cost of the dinner at each day. You have to find the minimal possible total cost of all dinners and the days when you should use flyers.

Input

The first line of the input contains n $(0 \le n \le 100)$ — the total number of days. Each of the next n lines contains the cost of the dinner at the corresponding day. A cost of a single dinner does not exceed 300.

Output

The first line of the output should contain the minimal possible total cost of all dinners. The second line should contain two integers k_1 and k_2 — the number of unused flyers after n days and the number of used flyers, respectively.

In the next k_2 lines output the identifier of the days in the increasing order when you should use a flyer. If there exist several solutions with the same minimal possible total cost, output any with the maximal k_1.

input

```
5
110
40
120
110
60
```

output

```
260
0 2
3
5
```

input

```
3
110
110
110
```

output

```
220
1 1
2
```

G. Removing brackets 2.0

2 seconds, 256 megabytes

Given a string made up of round, square and curly brackets. Find a subsequence of maximal length that forms the correct bracket sequence.

Input

The sole line of the input contains a string of round, square and curly brackets. The length of the string does not exceed 100 characters.

Output

The sole line of the output should contain the string with the maximum length that is a correct bracket sequence and can be obtained from the original string by deleting some characters.

input

```
([)]
```

output

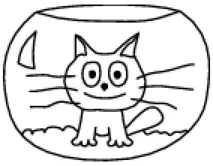
```
[]
```

H. The Salesman Problem

1 second, 256 megabytes

The Salesman (The Cat Aquariums Salesman) is going to visit n cities, visiting each city exactly once.

Help him to find the shortest possible path.



Input

The first line of the input contains one integer n ($1 \leqslant n \leqslant 13$) — the number of cities. Each of the next n lines contains n integers: the lengths of paths between cities.

The j-th integer in the i-th line is $a_{i,j}$ ($0 \leqslant a_{i,j} \leqslant 10^6$; $a_{i,j} = a_{j,i}$; $a_{i,i} = 0$) — the distance between cities i and j.

Output

The first line of the output should contain one integer — the smallest total distance that Salesman needs to travel. The second line should contain n integers — the order in which he should visit the cities.

input

```
5
0 183 163 173 181
183 0 165 172 171
163 165 0 189 302
173 172 189 0 167
181 171 302 167 0
```

output

```
666
4 5 2 3 1
```

I. Domino Tiling

1 second, 256 megabytes

You are given a rectangular grid of square cells. The character 'X' represents a cell that is already covered, the character '.' is a cell that still needs to be covered.

You want to cover all the '.' cells using a collection of disjoint 2×1 dominos. Return the number of ways to do this. Two ways are considered different if two cells are covered by the same domino in one tiling and by two different dominos in the other tiling.

input

```
2 2
..
..
```

output

```
2
```

input

```
3 3
...
.X.
...
```

output

```
2
```

input

```
3 3
...
...
..X
```

output

```
4
```

J. Cute Drawings

2 seconds, 256 megabytes

Find the number of colorings of a $n \times m$ grid in two colors without 2×2 squares of the same color.

Input

The first line of the input contains two integers n, m ($1 \le n \cdot m \le 30$) — the sizes of the grid.

Output

The sole line of the output should contain one integer: the number of colorings of a $n \times m$ grid in two colors without 2×2 squares of the same color. Colorings that can be made equal by rotations or reflections are considered different.

input

```
1 1
```

output

```
2
```

input

```
1 2
```

output

```
4
```

K. Cows in a Skyscraper

1 second, 256 megabytes

A little known fact about Bessie and friends is that they love stair climbing races. A better known fact is that cows really don't like going down stairs. So after the cows finish racing to the top of their favorite skyscraper, they had a problem. Refusing to climb back down using the stairs, the cows are forced to use the elevator in order to get back to the ground floor.

The elevator has a maximum weight capacity of w ($1 \le w \le 10^8$) pounds and cow i weighs c_i ($1 \le c_i \le w$) pounds. Please help Bessie figure out how to get all the n ($1 \le n \le 18$) of the cows to the ground floor using the least number of elevator rides. The sum of the weights of the cows on each elevator ride must be no larger than w.

Input

The first line of the input contains two integers n and w ($1 \le n \le 18$, $1 \le w \le 10^8$) — the number of cows and the maximum weight capacity of the elevator.

Next n lines describe weights of the cows: i-th line contains integer c_i ($1 \le c_i \le w$).

Output

The first line of the output should contain integer r — the minimum number of elevator rides needed.

Each of the next r lines should start with an integer giving the number of cows in the set, followed by the indices of the cows in the set during that ride.

input

```
4 10
5
6
3
7
```

output

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

There are four cows weighing 5, 6, 3, and 7 pounds. The elevator has a maximum weight capacity of 10 pounds.

We can put the cow weighing 3 on the same elevator as any other cow but the other three cows are too heavy to be combined. For the solution above, elevator ride 1 involves cow 1 and 3, elevator ride 2 involves cow 2, and elevator ride 3 involves cow 4. Several other solutions are possible for this input.
