Problem A. Transfusion

Input: standard input
Output: standard output
Time limit per test: 2 seconds
Memory limit per test: 256 megabytes

The Lady has n tanks. Each tank has a capacity of s liters. The Lady also has m bottles of water with volumes of a_1 liters, a_2 liters, …, a_m liters. Each bottle is completely filled with water.

The Lady can open any bottle and pour its contents into some of her tanks. But, if the tank becomes full at some point when the water is poured, then the Lady starts to get nervous and throws out the rest of the water in the bottle.

Help the Lady find out how many tanks can be completely filled with water?

Input

The first line contains integers n, s i m $(1 \le n, s, m \le 1000)$. The second line contains integers $a_1, a_2, \ldots, a_m$ $(1 \le a_i \le 3)$.

Output

Output one number - the number of tanks that can be completely filled with water.

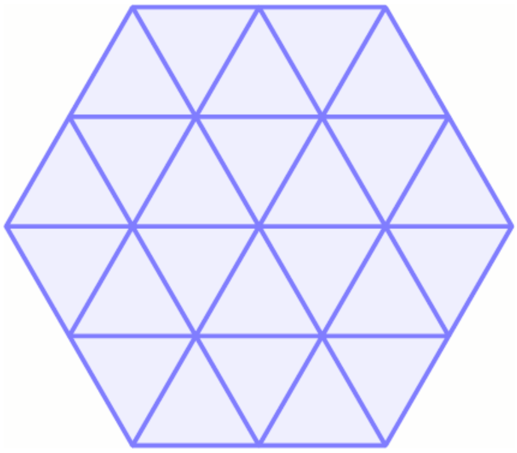
Examples

Input	Output
10 3 4 1 2 1 2	2
11 7 7 3 3 3 3 2 2 2	2

Problem B. Interesting dominoes

Input: standard input
Output: standard output
Time limit per test: 1 second
Memory limit per test: 128 megabyte

The Lady decided to play an interesting rhombic dominoes. For the game, she took a hexagon with side n, divided it into $6n^2$ proper triangles with side 1. Here is an example of such a hexagon with side 2.



An interesting rhombic dominoes consists of two proper triangles with side 1, adjacent on the side.

In how many ways can the Lady cover a hexagon with interesting rhombic dominoes (without overlays and extensions)?

Input

The first line contains a single number n $(1 \le n \le 7)$.

Output

Output one number – the amount of paving of the hexagon.

Examples

Input	Output
1	2

Problem C. Binary trees

Input: standard input
Output: standard output
Time limit per test: 1 seconds
Memory limit per test: 64 megabytes

There are two numbers ***N*** and ***H***. It is necessary to count the number of different binary search trees with ***N*** vertices, in which the maximum depth is not less than the number ***H.***

A binary search tree is a tree that has a key for each vertex – different numbers from ***1*** to ***N*.** And also at each vertex there are no more than two sons (left and right), and all the keys of the left subtree (if there are any) do not exceed the vertex key, and all keys of the right subtree (if there are any) exceed the vertex key. Maximum tree depth is the maximum number of vertices on the path from the root to any vertex of the tree (including the root and top).

Input

The first line contains two numbers ***N*** i ***H*** $(1 \le N, H \le 30)$.

Output

Output one number – the number of different binary search trees from ***N*** vertices, in which the maximum depth is not less than the number ***H***.

Examples

Input	Output
2 2	2
3 2	5

Problem D. Stumps

Input: standard input
Output: standard output
Time limit per test: 1 second
Memory limit per test: 64 megabytes

A track made of stumps was specially built for the Lady in the country, along which she must jump.

The track consists of N stumps of different heights. The Lady can jump a maximum of D centimeters up and any distance down. The Lady starts jumping to the left of the track and stands at zero height just before the first stump. The Lady's goal is to reach the other side of the track by jumping on each stump in order.

But, it may happen that the Lady will not be able to jump on some stumps because it will be too high.

Help the Lady cross the track by changing the heights of some stumps.

Calculate the minimum possible number of stumps that need to be increased or decreased so that the Lady can cross the track. The height of each of the stumps can be increased by any amount and reduced to an integral value.

Input

The first row contains two integers: number of stumps N $(1 \le N \le 200\ 000)$ and the maximum height M $(0 \le M \le 5\ 000)$, at which the Lady can jump. The next N rows contain the stumps heights A_i $(0 \le A_i \le 10^9)$, given as integers, one number per row.

Output

Output a single number – the minimum possible number of stumps, the height of which should be increased or decreased.

Examples

Input	Output
3 10 8 20 5	1

Problem E. The cherry festival

Input: standard input
Output: standard output
Time limit per test: 1 second
Memory limit per test: 64 megabytes

This year the cherry festival will be held on a rectangular field. The field has the form of a grid, where each cell is 1 square meter.

The organizers created a list of all G trees with cherries, and wrote down the coordinates of every tree. The festival will host a cherry picking competition with D participants. In order for the conditions for the participants to be fair, the field must be divided into D lanes in such a way that the maximum number of trees on one lane is minimal. The field can be divided only by vertical or horizontal lines that completely cut the field. The cuts should be made so that each cell of the field lies in exactly one lane. Each lane can be 0 in size.

Your task is to divide the field so that the maximum number of trees in one lane is minimal. You only need to output this number.

Input

The first row is given X, Y, D and G. They indicate the size of the field, the number of participants and the number of trees, respectively.

The next G rows contain two natural numbers x_i and y_i denoting the positions of the trees on the field.

Limitation: $2 \le X, Y \le 100\,000$, $1 \le G \le 100\,000$, $1 \le x_i \le X$, $1 \le y_i \le Y$, $2 \le D \le max(X, Y)$

Output

Output one number – the maximum number of trees in one lane in the optimal solution.

Examples

Input	Output
4 5 2 8 1 1 1 2 1 3 2 1 2 2 2 3 3 4 3 4	4

Problem F. Connecting cities

Input: standard input
Output: standard output
Time limit per test: 5 second
Memory limit per test: 64 megabytes

You found a map that depicted a specific country many centuries ago. There are only two cities on the map. These two cities are connected by a river. A river is a broken line that begins in one city and ends in another. Each straight segment of the river runs strictly from south to north, that is, in the direction of increasing y coordinate.

You became interested in this map and decided to experiment a bit. You want to build a road that will connect these two cities. The road, unfortunately, cannot easily cross the river: one bridge has to be built each time the road crosses the river. To build one meter of the road costs 1 coin, and to build one bridge costs T.

Find the minimum cost of the road between cities.

Input

The first row gives the number N, which indicates the number of turns of the river (including two cities) and the number T, which indicates the cost of one bridge. The next N rows contain two numbers X_i та Y_i $(1 \le i \le N)$. This pair indicates the coordinates (in meters) of the i-th turn of the river. These numbers satisfy the condition $Y_i < Y_{i+1}$, $1 \le i < N$. There are no three points lying on one straight line.

The road should start in (X_1, Y_1), and end in (X_n, Y_n).

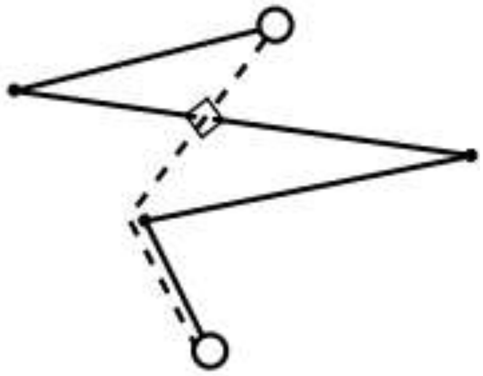
Limitation: For all tests $2 \le N \le 1500$, $0 < T \le 10^6$, $|X_i|$, $|Y_i| \le 10^5$.

Output

Your program should output a single number indicating the minimum cost of the road. The answer will be accepted if the relative error does not exceed 10^{-6}.

Examples

Input	Output
5 1 0 0 -1 2 4 3 -3 4 1 5	6.841619



Explanation to an example:

White circles indicate the cities where the road begins and ends. The black line is a river, the black dotted line is a road of minimal cost. The square marks the place where the bridge will need to be built.