Day 6

A. Power it!

2 seconds, 64 megabytes

You are given three ints: n, k, and m. For each i between 1 and n, inclusive, Fox Ciel calculated the number i^P, where P equals to $2^k - 1$. Find the sum of all numbers Fox Ciel calculated, modulo m

Input
Single line of input file contains three integers n, k and m. $1 \le n \le 10^6$, $1 \le k \le 400$, $2 \le m \le 10^9$.

Output
Output one integer — sum of all numbers Fox Ciel calculated, modulo m.

input
```
4 1 107
```
output
```
10
```

input
```
4 2 107
```
output
```
100
```

input
```
1000 400 1000000
```
output
```
750000
```

B. Private Party

2 seconds, 64 megabytes

Hero is inviting his friends to the party. He has n friends, numbered 0 through n-1. For each of his friends there is at most one other person the friend dislikes. You are given this information as array a with n elements. For each i, a[i] is either the number of the person disliked by friend i, we have a[i]=i if friend i likes everybody else.

Hero is inviting his friends one at a time. Whenever he invites friend i, they will accept if and only if the friend a[i] didn't accept an earlier invitation. (That includes two cases: either Hero didn't invite friend a[i] yet, or he did but the friend rejected the invitation.)

Hero noticed that the order in which he invites his friends matters: different orders may produce different numbers of accepted invitations. Find an order that will produce the most accepted invitations.

Input
First line contains one integer n — number of friends ($1 \le n \le 50$). Second line contains n integers — array a ($0 \le a[i] \le n - 1$)

Output
Output one integer — maximum number of accepted invitations.

input
```
2
0 1
```
output
```
2
```

input
```
2
1 0
```
output
```
1
```

input
```
4
1 0 3 2
```
output
```
2
```

input
```
6
5 2 2 4 5 0
```
output
```
5
```

C. Binary string 2

2 seconds, 64 megabytes

For two binary strings A and B we say that A is submask of B if:

1) They have the same length

2) For each position i where A[i] is 1, B[i] is also 1.

You are given two binary string s1 and s2 of same length. Count how many binary strings are submasks of at least one of s1 or s2.

Input
In the first line — first nonempty binary string with length no longer than 50 characters. In the second — second binary string with the same length.

Output
Output one integer — number of distinct strings with condition that string is either submask of s1 or s2 (or both).

input
```
01
10
```
output
```
3
```

input
```
111
111
```
output
```
8
```

input
```
110
011
```
output
```
6
```

input
```
11111111111111111111111111111111111111
00000000000000000000000000000000000000
```
output
```
274877906944
```

D. Hung Fu

2 seconds, 256 megabytes

The are two integer arrays a and b of length n. Consider the following formula:

$$\sum_{i=1}^{n} \min_{1 \le j \le i} a_i \oplus b_j.$$

You are practicing the calculation of the result of the above formula, and you have noticed that the order of elements in the arrays matters. Now you want to minimize the result of the calculation by permuting the elements of arrays a and b. More formally, you want to find such a permutation p that minimizes the following function:

$$F(p) = \sum_{i=1}^{n} \min_{1 \le j \le i} a_{p_i} \oplus b_{p_j}.$$

Find and output the lexicographically smallest permutation p that minimizes the function.

Input

On the first line, you are given a single integer n: the size of arrays $(1 \le n \le 50)$.

On the second line, you are given n integers a_i: the elements of array a $(0 \le a_i \le 1, 000, 000)$.

On the third line, you are given n integers b_i: the elements of array b $(0 \le b_i \le 1, 000, 000)$.

Output

On the first line, output a single integer: the minimum possible result of the function. On the second line, output n integers: the lexicographically smallest permutation p that minimizes the result of the function.

input
```
3
1 2 3
3 2 1
```
output
```
1
2 3 1
```

E. MaxMinTreeGame

2 seconds, 64 megabytes

MaxMinTreeGame is a game for two players. The game is played on a tree. The tree has N nodes, labeled 0 through N-1. Each node of the tree also has an integer cost. The players take alternating turns. In his turn, the current player starts by choosing one edge of the tree and erasing it. This necessarily divides the tree into two components. The current player then decides which of the components to keep, and erases the other component completely. The game ends when there is only one node left. The cost of that node is the result of the game.

The first player (i.e., the one that starts the game) wants to maximize the result. Naturally, the second player's goal is to minimize the result.

You are given tree and costs for each vertex. Find the result of the game, assuming that both players play optimally

Input

In the first line of input file a single integer — N. $(2 \le N \le 50)$

In the second line N-1 integers p[i] $(0 \le p[i] \le i)$ for each i from 0 to N - 1, there is an edge between vertices p[i] and i + 1(numbering starting from 0).

In the third line — N integers — costs for each vertex in order from vertex 0 to vertex N - 1. $(0 \le cost[i] \le 10^9)$

Output

Output one integer — result of the game.

input
```
2
0
4 6
```
output
```
6
```

input
```
3
0 1
4 6 5
```
output
```
5
```

input
```
5
0 1 2 3
0 1 0 1 0
```
output
```
0
```

input
```
3
0 0
3 2 5
```
output
```
5
```

F. Factory Repairs

4 seconds, 256 megabytes

A factory produces thimbles in bulk. Typically, it can produce up to a thimbles a day. However, some of the machinery is defective, so it can currently only produce b thimbles each day. The factory intends to choose a k-day period to do maintenance and construction; it cannot produce any thimbles during this time, but will be restored to its full production of a thimbles per day after the k days are complete.

Initially, no orders are pending. The factory receives updates of the form d_i, a_i, indicating that a_i new orders have been placed for the d_i-th day. Each order requires a single thimble to be produced on precisely the specified day. The factory may opt to fill as many or as few of the orders in a single batch as it likes.

As orders come in, the factory owner would like to know the maximum number of orders he will be able to fill if he starts repairs on a given day p_i. Help the owner answer his questions.

Input

The first line contains five integers n, k, a, b, and q $(1 \le k \le n \le 200\ 000,$ $1 \le b < a \le 10\ 000,$ $1 \le q \le 200\ 000)$ — the number of days, the length of the repair time, the production rates of the factory, and the number of updates, respectively.

The next q lines contain the descriptions of the queries. Each query is of one of the following two forms:

- $1\ d_i\ a_i$ $(1 \le d_i \le n,$ $1 \le a_i \le 10\ 000)$, representing an update of a_i orders on day d_i, or
- $2\ p_i$ $(1 \le p_i \le n - k + 1)$, representing a question: at the moment, how many orders could be filled if the factory decided to commence repairs on day p_i?

It's guaranteed that the input will contain at least one query of the second type.

Output

For each query of the second type, print a line containing a single integer — the maximum number of orders that the factory can fill over all n days.

input

```
5 2 2 1 8
1 1 2
1 5 3
1 2 1
2 2
1 4 2
1 3 2
2 1
2 3
```

output

```
3
6
4
```

input

```
5 4 10 1 6
1 1 5
1 5 5
1 3 2
1 5 2
2 1
2 2
```

output

```
7
1
```

Consider the first sample.

We produce up to 1 thimble a day currently and will produce up to 2 thimbles a day after repairs. Repairs take 2 days.

For the first question, we are able to fill 1 order on day 1, no orders on days 2 and 3 since we are repairing, no orders on day 4 since no thimbles have been ordered for that day, and 2 orders for day 5 since we are limited to our production capacity, for a total of 3 orders filled.

For the third question, we are able to fill 1 order on day 1, 1 order on day 2, and 2 orders on day 5, for a total of 4 orders.

G. Undiv2

2 seconds, 64 megabytes

Given a positive integer x, let s(x) be the second smallest positive integer that does not divide x. For example, let x = 6. The integers that do not divide x are 4, 5, 7, 8, 9, 10, ... The second smallest of these is 5. Hence we have s(6) = 5. Hero took a blank sheet of paper. For each i between 1 and n, inclusive, he computed the value s(i) and wrote it on the paper. You are given the int n. Compute and output the sum of the n numbers on Hero's paper.

Input

Input file contains a single positive number — n ($n \le 10^9$).

Output

Output a single integer — sum of numbers on Hero's paper.

input

```
1
```

output

```
3
```

input

```
2
```

output

```
7
```

input

```
3
```

output

```
11
```

input

```
123
```

output

```
523
```

H. Cycles

1 second, 256 megabytes

John Doe started thinking about graphs. After some thought he decided that he wants to paint an undirected graph, containing exactly k cycles of length 3.

A cycle of length 3 is an unordered group of three distinct graph vertices a, b and c, such that each pair of them is connected by a graph edge.

John has been painting for long, but he has not been a success. Help him find such graph. Note that the number of vertices there shouldn't exceed 100, or else John will have problems painting it.

Input

A single line contains an integer k ($1 \le k \le 10^5$) — the number of cycles of length 3 in the required graph.

Output

In the first line print integer n ($3 \le n \le 100$) — the number of vertices in the found graph. In each of next n lines print n characters "0" and "1": the i-th character of the j-th line should equal "0", if vertices i and j do not have an edge between them, otherwise it should equal "1". Note that as the required graph is undirected, the i-th character of the j-th line must equal the j-th character of the i-th line. The graph shouldn't contain self-loops, so the i-th character of the i-th line must equal "0" for all i.

input

```
1
```

output

```
3
011
101
110
```

input

```
10
```

output

```
5
01111
10111
11011
11101
11110
```

I. Telegram

time limit per test 3 seconds
memory limit per test 256 megabytes
input standard input
output standard output

There are N ($3 \le N \le 2 * 10^5$) students in Philip's class. Some pairs of them are friends to each other. There are exactly M ($2 \le M \le 2 * 10^5$) pairs of students in friendship.

Students need to spent C_{ij} ($1 \le C_{ij} \le 10^8$) time to support their friendship for each of the students in the pair, but teacher wants students to spend as less time as possible. So he asked Philip to break friendship between some pairs of students to have the smallest total sum of times for all students as possible, while keeping all students connected (so for each pair of students should exist some sequence of students so that adjacent students in this sequence are still friends). Philip also has his requirement, he want to left with exactly K ($1 \le K \le N$) friends.

Find the smallest total time students will spend for friendship while holding requirements above. It is guaranteed that all students are connected, but sometimes it could be impossible for Philip to left with K friends, in this case output -1.

Input

The first line contains three numbers N, M, K. The next M lines contain information about friends in format: $i\ j\ C_{ij}$. Where i, j are numbers of the students. Philip has a number 1.

Output

Output one number – minimal total time or -1 if it's impossible for Philip to have exactly K friends.

Examples

input

```
3 3 1
1 2 2
1 3 3
2 3 10
```

output

```
24
```

input

```
5 6 2
1 2 2
1 3 3
2 3 15
1 4 7
4 5 8
5 3 5
```

output

```
36
```

input

```
5 5 1
1 2 4
2 3 6
2 4 10
3 4 8
1 5 3
```

output

```
-1
```