CODEFORCES β
Sponsored by Telegram

15-295 Spring 2017 #7

A. Put Knight! (1/2 point)

time limit per test: 1 second
memory limit per test: 256 megabytes
input: input.txt
output: output.txt

Petya and Gena play a very interesting game "Put a Knight!" on a chessboard $n \times n$ in size. In this game they take turns to put chess pieces called "knights" on the board so that no two knights could threat each other. A knight located in square (r, c) can threat squares $(r - 1, c + 2)$, $(r - 1, c - 2)$, $(r + 1, c + 2)$, $(r + 1, c - 2)$, $(r - 2, c + 1)$, $(r - 2, c - 1)$, $(r + 2, c + 1)$ and $(r + 2, c - 1)$ (some of the squares may be located outside the chessboard). The player who can't put a new knight during his move loses. Determine which player wins considering that both players play optimally well and Petya starts.

Input

The first line contains integer T $(1 \le T \le 100)$ — the number of boards, for which you should determine the winning player. Next T lines contain T integers n_i $(1 \le n_i \le 10000)$ — the sizes of the chessboards.

Output

For each $n_i \times n_i$ board print on a single line "0" if Petya wins considering both players play optimally well. Otherwise, print "1".

Examples

input

```
2
2
1
```

output

```
1
0
```

B. Bits (1 point)

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Let's denote as $\text{popcount}(x)$ the number of bits set ('1' bits) in the binary representation of the non-negative integer x.

You are given multiple queries consisting of pairs of integers l and r. For each query, find the x, such that $l \le x \le r$, and $\text{popcount}(x)$ is maximum possible. If there are multiple such numbers find the smallest of them.

Input

The first line contains integer n — the number of queries ($1 \le n \le 10000$).

Each of the following n lines contain two integers l_i, r_i — the arguments for the corresponding query ($0 \le l_i \le r_i \le 10^{18}$).

Output

For each query print the answer in a separate line.

Examples

input

```
3
1 2
2 4
1 10
```

output

```
1
3
7
```

Note

The binary representations of numbers from 1 to 10 are listed below:

$1_{10} = 1_2$

$2_{10} = 10_2$

$3_{10} = 11_2$

$4_{10} = 100_2$

$5_{10} = 101_2$

$6_{10} = 110_2$

$7_{10} = 111_2$

$8_{10} = 1000_2$

$9_{10} = 1001_2$

$10_{10} = 1010_2$

C. Little Girl and Game (1 point)

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

The Little Girl loves problems on games very much. Here's one of them.

Two players have got a string s, consisting of lowercase English letters. They play a game that is described by the following rules:

- The players move in turns; In one move the player can remove an arbitrary letter from string s.
- If the player before his turn can reorder the letters in string s so as to get a palindrome, this player wins. A palindrome is a string that reads the same both ways (from left to right, and vice versa). For example, string "abba" is a palindrome and string "abc" isn't.

Determine which player will win, provided that both sides play optimally well — the one who moves first or the one who moves second.

Input

The input contains a single line, containing string s ($1 \leq |s| \leq 10^3$). String s consists of lowercase English letters.

Output

In a single line print word "First" if the first player wins (provided that both players play optimally well). Otherwise, print word "Second". Print the words without the quotes.

Examples

input
```
aba
```
output
```
First
```

input
```
abca
```
output
```
Second
```

D. Interesting Game (2 points)

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Two best friends Serozha and Gena play a game.

Initially there is one pile consisting of n stones on the table. During one move one pile should be taken and divided into an arbitrary number of piles consisting of $a_1 > a_2 > ... > a_k > 0$ stones. The piles should meet the condition $a_1 - a_2 = a_2 - a_3 = ... = a_{k-1} - a_k = 1$. Naturally, the number of piles k should be no less than two.

The friends play in turns. The player who cannot make a move loses. Serozha makes the first move. Who will win if both players play in the optimal way?

Input

The single line contains a single integer n ($1 \le n \le 10^5$).

Output

If Serozha wins, print k, which represents the minimal number of piles into which he can split the initial one during the first move in order to win the game.

If Gena wins, print "-1" (without the quotes).

Examples

input
3
output
2

input
6
output
-1

input
100
output
8

E. Game of Stones (2 points)

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

Sam has been teaching Jon the *Game of Stones* to sharpen his mind and help him devise a strategy to fight the white walkers. The rules of this game are quite simple:

- The game starts with n piles of stones indexed from 1 to n. The i-th pile contains s_i stones.
- The players make their moves alternatively. A move is considered as removal of some number of stones from a pile. Removal of 0 stones does not count as a move.
- The player who is unable to make a move loses.

Now Jon believes that he is ready for battle, but Sam does not think so. To prove his argument, Sam suggested that they play a modified version of the game.

In this modified version, no move can be made more than once on a pile. For example, if 4 stones are removed from a pile, 4 stones cannot be removed from that pile again.

Sam sets up the game and makes the first move. Jon believes that Sam is just trying to prevent him from going to battle. Jon wants to know if he can win if both play optimally.

Input

First line consists of a single integer n $(1 \le n \le 10^6)$ — the number of piles.

Each of next n lines contains an integer s_i $(1 \le s_i \le 60)$ — the number of stones in i-th pile.

Output

Print a single line containing "YES" (without quotes) if Jon wins, otherwise print "NO" (without quotes)

Examples

input
```
1
5
```
output
```
NO
```

input
```
2
1
2
```
output
```
YES
```

Note

In the first case, Sam removes all the stones and Jon loses.

In second case, the following moves are possible by Sam: $\{1,2\} \to \{0,2\}, \{1,2\} \to \{1,0\}, \{1,2\} \to \{1,1\}$

In each of these cases, last move can be made by Jon to win the game as follows: $\{0,2\} \to \{0,0\}, \{1,0\} \to \{0,0\}, \{1,1\} \to \{0,1\}$

F. Permutation Sum **(2 points)**

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

Permutation *p is an ordered set of integers $p_1, p_2, \ldots, p_n$, consisting of n distinct positive integers, each of them doesn't exceed n. We'll denote the i-th element of permutation p as p_i. We'll call number n the size or the length of permutation $p_1, p_2, \ldots, p_n$.*

Petya decided to introduce the sum operation on the set of permutations of length n. Let's assume that we are given two permutations of length n: $a_1, a_2, \ldots, a_n$ and $b_1, b_2, \ldots, b_n$. Petya calls the sum of permutations a and b such permutation c of length n, where $c_i = ((a_i - 1 + b_i - 1) \ mod \ n) + 1 \ (1 \le i \le n)$.

Operation $x \mod y$ means taking the remainder after dividing number x by number y.

Obviously, not for all permutations a and b exists permutation c that is sum of a and b. That's why Petya got sad and asked you to do the following: given n, count the number of such pairs of permutations a and b of length n, that exists permutation c that is sum of a and b. The pair of permutations x, y $(x \ne y)$ and the pair of permutations y, x are considered distinct pairs.

As the answer can be rather large, print the remainder after dividing it by 1000000007 $(10^9 + 7)$.

Input

The single line contains integer n $(1 \le n \le 16)$.

Output

In the single line print a single non-negative integer — the number of such pairs of permutations a and b, that exists permutation c that is sum of a and b, modulo 1000000007 $(10^9 + 7)$.

Examples

input
3
output
18

input
5
output
1800

G. Remembering Strings (2 points)

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

You have multiset of n strings of the same length, consisting of lowercase English letters. We will say that those strings are easy to remember if for each string there is some position i and some letter c of the English alphabet, such that this string is the only string in the multiset that has letter c in position i.

For example, a multiset of strings {"abc", "aba", "adc", "ada"} are not easy to remember. And multiset {"abc", "ada", "ssa"} is easy to remember because:

- the first string is the only string that has character c in position 3;
- the second string is the only string that has character d in position 2;
- the third string is the only string that has character s in position 2.

You want to change your multiset a little so that it is easy to remember. For a_{ij} coins, you can change character in the j-th position of the i-th string into any other lowercase letter of the English alphabet. Find what is the minimum sum you should pay in order to make the multiset of strings easy to remember.

Input

The first line contains two integers n, m $(1 \le n, m \le 20)$ — the number of strings in the multiset and the length of the strings respectively. Next n lines contain the strings of the multiset, consisting only of lowercase English letters, each string's length is m.

Next n lines contain m integers each, the i-th of them contains integers $a_{i1}, a_{i2}, ..., a_{im}$ $(0 \le a_{ij} \le 10^6)$.

Output

Print a single number — the answer to the problem.

Examples

input

```
4 5
abcde
abcde
abcde
abcde
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1
```

output

```
3
```

input

```
4 3
abc
aba
adc
ada
10 10 10
10 1 10
10 10 10
10 1 10
```

output

```
2
```

input

```
3 3
abc
ada
ssa
1 1 1
1 1 1
1 1 1
```

output

```
0
```

H. An easy problem about trees **(3 points)**

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Pieguy and Piegirl are playing a game. They have a rooted binary tree, that has a property that each node is either a leaf or has exactly two children. Each leaf has a number associated with it.

On his/her turn a player can choose any two leafs that share their immediate parent, remove them, and associate either of their values with their parent, that now became a leaf (the player decides which of the two values to associate). The game ends when only one node (the one that was the root of the tree) is left.

Pieguy goes first, and his goal is to maximize the value that will be associated with the root when the game ends. Piegirl wants to minimize that value. Assuming that both players are playing optimally, what number will be associated with the root when the game ends?

Input

First line contains a single integer t $(1 \le t \le 100)$ — number of test cases. Then t test cases follow. Each test case begins with an empty line, followed by a line with a single integer n $(1 \le n \le 250)$, followed by n lines describing n nodes of the tree. Each of those n lines either contains a non-negative number a_i, indicating a leaf node with value a_i $(0 \le a_i \le 1000)$ associated with it, or -1 followed by integers l and r, indicating a non-leaf node with children l and r $(0 \le l, r \le n - 1)$. Nodes are numbered from 0 to $n - 1$. The root is always node 0.

Output

For each test case print one line with one integer on it — the number that will be associated with the root when the game ends.

Examples

input

```
4

3
-1 1 2
10
5

5
-1 1 2
-1 3 4
10
5
20

7
-1 1 2
-1 3 4
-1 5 6
1
2
3
4

11
-1 1 2
-1 3 4
-1 5 6
-1 7 8
15
7
-1 9 10
7
8
9
11
```

output

```
10
10
4
8
```