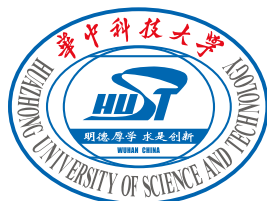


# 2022 Huazhong University of Science and Technology "Freshman Cup" Programming Contest

HUST ACM-ICPC Team

Sponsored by Huawei Technologies Co., Ltd.

Oct 22, 2022



華中科技大學  
HUAZHONG UNIVERSITY OF SCIENCE AND TECHNOLOGY



## Problems

- A. Monster Killer
- B. Subset Counting
- C. Roll the Circle
- D. Random
- E. Merging Stones
- F. String Problem
- G. Math Learning
- H. Substrings
- I. Photos
- J. Two Kings
- K. The Grandstand
- L. Phigros
- M. Walk Alone's Conjecture
- N. Coloring

## Problem A. Monster Killer

Input file:           standard input  
Output file:         standard output  
Time limit:          1 second  
Memory limit:       256 megabytes

Walk Alone is playing a game recently. In this game, there are  $n$  monsters numbered from 1 to  $n$ , the  $i$ -th of which has an attack ability of  $a_i$ . Walk Alone is required to kill all of them in order.

To kill a monster, he needs to battle with it. He can battle with the  $(i + 1)$ -th monster if and only if all monsters from 1 to  $i$  are killed.

Initially, Walk Alone's attack ability  $m$  equals to 0. After a battle with the  $i$ -th monster, three situations may happen:

- $m = a_i$ , he kills the monster.
- $m > a_i$ , he kills the monster, but his attack ability  $m$  may become  $m - 1$  with a probability of  $p_i$ .
- $m < a_i$ , he loses the battle. The monster is still alive, but his attack ability  $m$  may become  $m + 1$  with a probability of  $p_i$ .

When he loses a battle, he will keep trying until he wins.

Now Walk Alone wants to know the **expected number of battles** he needs to go through in total.

It can be shown that the answer can always be expressed as an irreducible fraction  $x/y$ , where  $x$  and  $y$  are integers and  $y \not\equiv 0 \pmod{998\,244\,353}$ . Output such an integer  $a$  that  $0 \leq a < 998\,244\,353$  and  $a \cdot y \equiv x \pmod{998\,244\,353}$ .

### Input

The first line contains an integer  $n$  ( $1 \leq n \leq 10^3$ ), the number of monsters.

The  $i$ -th of the next  $n$  lines contains two integers  $a_i$  ( $0 \leq a_i \leq 10^3$ ) and  $x_i$  ( $1 \leq x_i \leq 100$ ), and  $p_i = 0.01 x_i$ .

### Output

Output a single number, the expected number of battles modulo 998 244 353.

### Examples

| standard input                            | standard output |
|-------------------------------------------|-----------------|
| 3<br>1 100<br>0 50<br>1 100               | 499122181       |
| 5<br>2 50<br>6 30<br>1 20<br>5 50<br>4 40 | 332748140       |

### Note

In the first example, the result of each battle is as below:

- Battle 1: monster 1 is still alive, and his attack ability become 1.

- Battle 2: monster 1 is killed, and his attack ability remains 1.
- Battle 3: monster 2 is killed, and with 50% probability his attack ability become 0.
- If his attack ability is 0:
  - Battle 4: monster 3 is still alive, and his attack ability become 1.
  - Battle 5: monster 3 is killed.

If his attack ability is 1, monster 3 is killed in battle 4.

The expected number of battles is 4.5, which is congruent to 499 122 181 modulo 998 244 353.

## Problem B. Subset Counting

Input file:            **standard input**  
Output file:         **standard output**  
Time limit:          2 seconds  
Memory limit:       512 megabytes

Given three integers  $n, k$  and  $m$ . Let  $S = \{i \mid 1 \leq i \leq nm + k, i \in \mathbb{Z}\}$ . For all integers  $r$  where  $0 \leq r < m$ , you need to output the number of sets  $T$  such that

- $T$  is a subset of  $S$ .
- The sum of elements in  $T$  is modulo  $m$  exactly  $r$ .

### Input

The only line of the input contains three integers  $n$  ( $0 \leq n \leq 10^{13}$ ),  $k$  ( $0 \leq k < \min(m, 500)$ ,  $nm + k > 0$ ) and  $m$  ( $1 \leq m \leq 10^5$ ) as described above.

### Output

Output  $m$  integers in a line, the  $i$ -th of which indicates the number of subsets modulo 998 244 353 when  $r = i - 1$ .

### Examples

| standard input | standard output                                                                                              |
|----------------|--------------------------------------------------------------------------------------------------------------|
| 1 1 2          | 4 4                                                                                                          |
| 1919 8 10      | 577613260 577613260 822345879<br>577613260 822345879 577613260<br>577613260 822345879 577613260<br>822345879 |

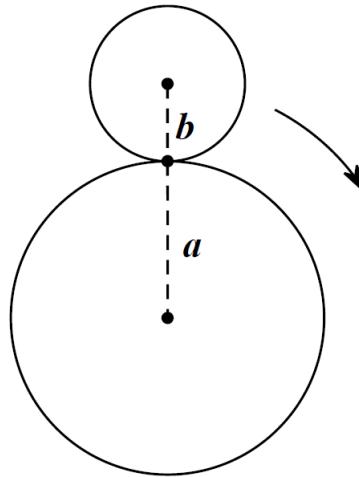
### Note

The output of the second example is wrapped due to limited page width.

## Problem C. Roll the Circle

Input file:           standard input  
Output file:         standard output  
Time limit:          1 second  
Memory limit:       256 megabytes

Walk Alone has two circles. The two circles are externally tangent, and are of radius  $a$  and  $b$  respectively. He keeps the first circle of radius  $a$  stationary, and rotates the second one around the first one, keeping the point of contact without slipping. He wants to know how many times will the second circle revolve after rolling it exactly one trip back to its starting point.



### Input

The input consists of multiple test cases.

The first line contains an integer  $T$  ( $1 \leq T \leq 10^5$ ), the number of test cases.

For each test case, the first and the only line contains two integers  $a$  and  $b$  ( $1 \leq a, b \leq 10^{18}$ ), denoting the radius of two circles.

### Output

For each test case output the answer in a line. The answer should be expressed as a minimal fraction  $p/q$  where  $\gcd(p, q) = 1$ .

### Example

| standard input | standard output |
|----------------|-----------------|
| 2              | 3/1             |
| 2 1            | 5/3             |
| 4 6            |                 |

## Problem D. Random

Input file:            **standard input**  
Output file:          **standard output**  
Time limit:           1 second  
Memory limit:        256 megabytes

Walk Alone has a special random number generator, the pseudocode of which is as follows:

```
uintk rand(uintk x, int A[]) {  
    for (int i = 0; i < A.size(); ++i) {  
        if (A[i] >= 0) x ^= x << A[i];  
        else x ^= x >> -A[i];  
    }  
    return x;  
}
```

Here ‘uintk’ denotes the type of unsigned integers with  $k$  binary bits (‘bitset<k>’ in C++).

Given an array  $A$  and an integer  $k$ , Walk Alone wants to know the number of integer  $x$  ( $0 \leq x < 2^k$ ) such that  $\text{rand}(x, A) = x$ .

### Input

The first line contains two integers  $n$  ( $0 \leq n \leq 10^3$ ) and  $k$  ( $1 \leq k \leq 10^3$ ), the size of array  $A$  and the length of type ‘uintk’ respectively.

The second line contains  $n$  integers, the  $i$ -th of which is  $A_i$  ( $|A_i| < k$ ).

### Output

Print the number of  $x$  modulo 998 244 353.

### Examples

| standard input | standard output |
|----------------|-----------------|
| 3 5<br>2 3 -4  | 2               |
| 0 3            | 8               |

## Problem E. Merging Stones

Input file:            `standard input`  
Output file:        `standard output`  
Time limit:        1 second  
Memory limit:     256 megabytes

Walk Alone has  $n$  piles of stones, and there are  $a_i$  stones in the  $i$ -th pile.

He can do an operation to these stones for  $n - 1$  times. In each operation, he can select two piles of stones of size  $x$  and  $y$ , and merge them (the two piles will become a larger pile of size  $x + y$ ), and he will get  $x \cdot y$  score after that.

Walk Alone wants to know the maximal total score he can get. Can you help him?

### Input

The first line contains an integer  $n$  ( $1 \leq n \leq 10^5$ ), the number of piles of stones.

The second line contains  $n$  integers. The  $i$ -th integer  $a_i$  ( $1 \leq a_i \leq 10^4$ ) indicates the number of stones in the  $i$ -th pile.

### Output

Output an integer representing the maximal score Walk Alone can get.

### Examples

| standard input   | standard output |
|------------------|-----------------|
| 3<br>1 2 3       | 11              |
| 6<br>1 1 4 5 1 4 | 98              |

## Problem F. String Problem

Input file:           standard input  
Output file:         standard output  
Time limit:          3 seconds  
Memory limit:       512 megabytes

Walk Alone loves strings, and he insists that a contest without a string problem is not a good contest, so here is the string problem.

You are given two strings  $s$  and  $t$ . You need to select the longest continuous substring  $s'$  from  $s$ , such that it is possible to reverse at most one prefix of  $t$ , and after that there exists a prefix of  $t$  (not necessarily the same with the previous one) that equals to  $s'$ .

### Input

The first line contains two integers  $n$  ( $1 \leq n \leq 2 \cdot 10^5$ ) and  $m$  ( $1 \leq m \leq 2 \cdot 10^5$ ), indicating the length of  $s$  and  $t$ .

The second line and the third line are two strings consisting of only lowercase characters, representing string  $s$  and  $t$ , respectively.

### Output

Output the length of the longest substring of  $s$  in a line.

### Examples

| standard input             | standard output |
|----------------------------|-----------------|
| 8 7<br>cacbabca<br>abcabcc | 6               |
| 5 5<br>abcde<br>fdcba      | 4               |

### Note

In the first example, you can reverse the prefix of length 4 of  $t$ . After that  $t$  becomes 'acbabcc', whose prefix of length 6 coincides with  $s_{1:7}$ , i.e., 'acbabca'.

## Problem G. Math Learning

Input file:            **standard input**  
Output file:         **standard output**  
Time limit:          4 seconds  
Memory limit:       256 megabytes

Walk Alone is afraid of math, especially memorizing formulas. In Walk Alone's math textbook, there are  $n$  formulas connected with  $n - 1$  edges, which form a tree rooted at the first formula. To handle the  $i$ -th formula, he must remember all formulas in its subtree, including the formula itself, and memorizing the  $i$ -th formula consumes  $h_i$  energy.

One day after math class, Walk Alone is assigned with  $m$  math problems as his homework, and he should complete them in order. To complete the  $i$ -th problem, he needs to handle the  $x_i$ -th formula. However, when he is solving the  $i$ -th problem, he can only remember all formulas used in the recent  $k_i$  problems. Formally speaking, he remembers formula  $u$  if and only if there exists a positive integer  $p$  such that  $i - k_i \leq p < i$  and  $u$  is in the subtree of  $x_p$ . For all formulas in the subtree of  $x_i$  that he does not remember, he will consume energy to memorize them.

Walk Alone wants to know the amount of energy he should consume to finish each problem.

### Input

The first line contains two integers  $n$  ( $1 \leq n \leq 10^6$ ) and  $m$  ( $1 \leq m \leq 2 \cdot 10^5$ ), indicating the number of formulas and problems, respectively.

The second line contains  $n$  integers, the  $i$ -th of which,  $h_i$  ( $1 \leq h_i \leq 10^9$ ), indicates the energy needed to memorize the  $i$ -th formula.

Each of the following  $n - 1$  lines contains two integers  $u$  and  $v$  ( $1 \leq u, v \leq n$ ), indicating an edge between  $u$  and  $v$ .

The  $i$ -th of the following  $m$  lines contains two integers  $x_i$  ( $1 \leq x_i \leq n$ ) and  $k_i$  ( $0 \leq k_i \leq m$ ) as described above.

### Output

Output the amount of energy Walk Alone needs to consume to finish each problem.

### Example

| standard input | standard output |
|----------------|-----------------|
| 5 5            | 31              |
| 1 2 4 8 16     | 0               |
| 1 2            | 0               |
| 1 3            | 16              |
| 2 4            | 7               |
| 2 5            |                 |
| 1 1            |                 |
| 2 1            |                 |
| 4 1            |                 |
| 5 1            |                 |
| 1 2            |                 |

## Problem H. Substrings

Input file:            **standard input**  
Output file:         **standard output**  
Time limit:          1 second  
Memory limit:       256 megabytes

Walk Alone has a string, and he wants you to count the number of **consecutive** 'fyt' in the string.

For example, the answer to 'ffyt' is 1 since the second to the fourth character forms a consecutive 'fyt', while 'ffyytt' has no 'fyt' as its substring.

### Input

The first line contains an integer  $n$  ( $1 \leq n \leq 10^6$ ), indicating the length of the string.

The second line contains a single string consisting of three characters 'f', 'y', and 't', representing the string.

### Output

Output a single integer indicating the number of consecutive 'fyt' in the input string.

### Examples

| standard input  | standard output |
|-----------------|-----------------|
| 6<br>fytfyt     | 2               |
| 9<br>fyttffyytt | 1               |

## Problem I. Photos

Input file:            **standard input**  
Output file:          **standard output**  
Time limit:           2 seconds  
Memory limit:        256 megabytes

ZCY and YYH are good friends. One day, ZCY took a photo of YYH eating a hamburger, and she wants to stick it on every building in the school.

The school contains  $n$  buildings connected with  $n - 1$  roads, and there is exactly one path between any two buildings. Initially, ZCY is at building  $a$ , and YYH is at building  $b$ . In every minute, both of them can walk along a road to the building on the other side. ZCY wants to visit as many buildings as she can, so that she can stick YYH's photos on them. Of course YYH does not want her photo to be found on too many building, so she will try to catch ZCY and stop her as soon as possible. At any moment when YYH and ZCY are at the same building or on the same road, ZCY will be caught by YYH.

Now ZCY wants you to tell her the maximum number of photos that she can stick on the buildings.

### Input

The first line contains three integers  $n$  ( $1 \leq n \leq 10^6$ ),  $a$  and  $b$  ( $1 \leq a, b \leq n$ ) as described above.

Each of the following  $n - 1$  lines contains two integers  $u$  and  $v$  ( $1 \leq u, v \leq n$ ), indicating a road between  $u$  and  $v$ .

### Output

Output an integer indicating the maximum number of buildings ZCY can visit.

### Examples

| standard input                    | standard output |
|-----------------------------------|-----------------|
| 5 4 3<br>1 2<br>1 3<br>2 4<br>2 5 | 3               |
| 3 2 2<br>1 2<br>2 3               | 0               |

### Note

In the first example, the buildings that ZCY visit are  $4 \rightarrow 2 \rightarrow 5$  and YYH are  $3 \rightarrow 1 \rightarrow 2$ . ZCY will be caught by YYH at building 5 eventually.

## Problem J. Two Kings

Input file:            **standard input**  
Output file:         **standard output**  
Time limit:          1 second  
Memory limit:       256 megabytes

Walk Alone and Salix Leaf are playing a special chess game on an infinite chess board, the center of which locates at  $(0,0)$ . There are two Kings on the chess board, the white one belongs to Walk Alone and the black one belongs to Salix Leaf. Walk Alone moves first, after which they take turns moving their Kings.

In one move, a King moves one square in any direction. Formally speaking, if the king locates at  $(x, y)$  at the beginning of one's turn, then it must move to one of  $(x + \Delta x, y + \Delta y)$  where  $\max(|\Delta x|, |\Delta y|) = 1$ . A King must move if it is able to move. Additionally, a King is forbidden to move into the nearest 8 blocks of the other King, or the other King may capture it during the next move.

Walk Alone wants to keep moving right, which means if Walk Alone moves to  $(10^{100}, y)$  for some integer  $y$ , he wins. Conversely, Salix Leaf wants to stop him, which means if it is impossible for Walk Alone to move to the place he wants, Salix Leaf wins.

Now the initial coordinates of the two Kings are given, and you should decide who will win if both of them play optimally. If Walk Alone will win, print 'Walk Alone'; otherwise, print 'Salix Leaf' (without quotes).

### Input

The input contains multiple test cases.

The first line of the input contains an integer  $T$  ( $1 \leq T \leq 10^5$ ), the number of test cases.

Each test case contains only one line with four integers  $x_1, y_1, x_2, y_2$ . The coordinates of Walk Alone's and Salix Leaf's Kings are  $(x_1, y_1)$  and  $(x_2, y_2)$ , respectively. It is guaranteed that  $\max(|x_1 - x_2|, |y_1 - y_2|) > 1$ .

The absolute value of all coordinates are less or equal to  $10^9$ .

### Output

For each test case, output the winner in one line.

### Example

| standard input | standard output |
|----------------|-----------------|
| 3              | Walk Alone      |
| 0 -1 2 1       | Salix Leaf      |
| -2 3 2 3       | Walk Alone      |
| 2 0 -1 0       |                 |

## Problem K. The Grandstand

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

On the 70th anniversary of HUST (Hualishu University of Science and Technology), the grandstand in the playground near QinYuan is renewed!

Walk Alone loves the new grandstand. He finds that the new grandstand is a matrix of  $n \times m$  seats with  $n$  rows and  $m$  columns, and each seat is colored either blue, red or white. The red seats in the grandstand form the shape of 'HUST' together.

What is more, Walk Alone finds that the shape of the four letters should satisfy the following rules:

- 'H'. There exists positive integers  $x_0, y_0, x, y, z, d$ , such that

$$R = [0, 0, x + y + d, d] \cup [0, z + d, x + y + d, z + 2d] \cup [x, d, x + d, z + d]$$

where  $[x_1, y_1, x_2, y_2] = \{(x + x_0, y + y_0) \mid x_1 \leq x < x_2, y_1 \leq y < y_2\}$ , and  $R$  denotes the set of coordinates of all red seats.

- 'U'. There exists positive integers  $x_0, y_0, x, y, d$ , such that

$$R = [0, 0, x + d, y + 2d] - [0, d, x, y + d]$$

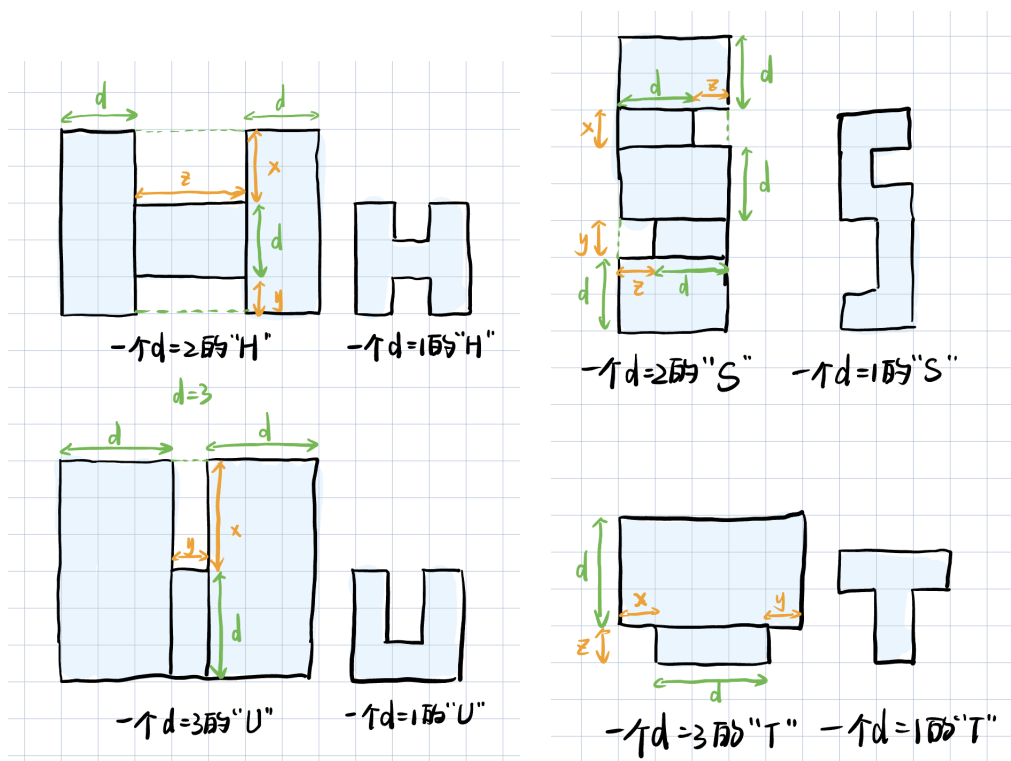
- 'S'. There exists positive integers  $x_0, y_0, x, y, z, d$ , such that

$$R = [0, 0, x + y + 3d, z + d] - [d, d, x + d, z + d] - [x + 2d, 0, x + y + 2d, z]$$

- 'T'. There exists positive integers  $x_0, y_0, x, y, z, d$ , such that

$$R = [0, 0, d, x + y + d] \cup [d, x, z + d, x + d]$$

For better understanding, it is recommended to see the illustration below ( $x, y, z$  and  $d$  are arbitrary positive integers):



Now Walk Alone gives you a part of the grandstand, in which some seats are colored red. You need recognize whether it belongs to one of 'H', 'U', 'S', 'T', or Walk Alone is cheating (it does not belong to any of the four letters).

## Input

The input contains multiple test cases.

The first line of the input contains an integer  $T$  ( $1 \leq T \leq 10$ ), the number of test cases.

For each test case, the first line contains two integers  $n$  and  $m$  ( $1 \leq n, m \leq 3000$ ), indicating the size of grandstand.

Each of the next  $n$  lines is a string of length  $m$  consisting only of '.' and 'x'. The  $j$ -th character in the  $i$ -th line is 'x' if and only if the seat located at the  $j$ -th column and the  $i$ -th row is red.

## Output

For each test case, output one of 'H', 'U', 'S' or 'T' as the answer, or 'OOPS!' (without quotes) if none of them are correct.

## Example

| standard input | standard output |
|----------------|-----------------|
| 4              | S               |
| 6 5            | H               |
| .xxx.          | OOPS!           |
| .x...          | OOPS!           |
| .xxx.          |                 |
| ...x.          |                 |
| ...x.          |                 |
| .xxx.          |                 |
| 5 5            |                 |
| xx.xx          |                 |
| xx.xx          |                 |
| xxxxx          |                 |
| xxxxx          |                 |
| xx.xx          |                 |
| 5 5            |                 |
| xx..x          |                 |
| xx..x          |                 |
| xxxxx          |                 |
| xxxxx          |                 |
| xx..x          |                 |
| 2 3            |                 |
| xxx            |                 |
| ..x            |                 |

## Problem L. Phigros

Input file:            **standard input**  
Output file:          **standard output**  
Time limit:           1 second  
Memory limit:        256 megabytes

Phigros is a music game in which players are required to touch the screen following the rhythm of songs and try to get the highest scores.

The total score of a song is the sum of the following two parts **rounded to the nearest integer** (assume that the song has  $n$  notes in total):

- Accuracy. The accuracy score for each note is added up independently according to the judgment:
  - Perfect:  $900\,000 \cdot n^{-1}$ ;
  - Good:  $585\,000 \cdot n^{-1}$ ;
  - Bad or miss: 0.
- Combo. The Max Combo of a song is the maximum number of **continuous** notes with **perfect or good** judgements. The combo score for a song with Max Combo  $m$  is  $100\,000 m \cdot n^{-1}$ .

Recall that

$$\text{round}(x) = \begin{cases} \lfloor x \rfloor, & x < \lfloor x \rfloor + 0.5 \\ \lfloor x \rfloor + 1, & x \geq \lfloor x \rfloor + 0.5 \end{cases}$$

Now Walk Alone is playing a song with  $n$  notes. He has already got the judgement of the first  $k$  notes, and he wants to know the maximum score he could get if he played the rest of the song perfectly.

### Input

The first line contains two integers  $n$  and  $k$  ( $1 \leq k \leq n \leq 4000$ ), the note count of the song and the number of notes Walk Alone has played.

The  $i$ -th of the following  $k$  lines contains one of ‘perfect’, ‘good’, ‘bad’ and ‘miss’ (without quotes), indicating the judgement of the  $i$ -th note.

### Output

An integer indicating the maximum score Walk Alone could get.

### Examples

| standard input                                   | standard output |
|--------------------------------------------------|-----------------|
| 6 5<br>perfect<br>good<br>perfect<br>bad<br>good | 695000          |
| 1000 2<br>miss<br>good                           | 998685          |

### Note

A song has at most 4000 notes in Phigros.

## Problem M. Walk Alone's Conjecture

Input file:            standard input  
Output file:           standard output  
Time limit:           1 second  
Memory limit:        256 megabytes

Walk Alone has a conjecture which says

- Any positive integer can be expressed as the difference between two positive integers with the same number of prime factors.

He fails to prove it, so he wants you to prove it in a small range. Your work is to construct two integers  $x$  and  $y$  such that

- $y - x = n$ .
- $x$  and  $y$  has the same number of prime factors.

### Input

The first line contains one integer  $T$  ( $1 \leq T \leq 10^5$ ), the number of test cases.

Each test case consists of only one line with an integer  $n$  ( $1 \leq n \leq 10^8$ ), denoting the number Walk Alone wants you to prove.

### Output

For each test case, print two integers  $x$  and  $y$  ( $1 \leq x < y \leq 10^{10}$ ) in one line satisfying the constraints above.

### Example

| standard input | standard output |
|----------------|-----------------|
| 3              | 2 3             |
| 1              | 2 4             |
| 2              | 12 15           |
| 3              |                 |

## Problem N. Coloring

Input file:            **standard input**  
Output file:          **standard output**  
Time limit:           1 second  
Memory limit:        256 megabytes

Walk Alone is poor at graph theory. In today's graph theory class, Walk Alone is given a graph with  $n$  vertices and  $m$  edges, and he is required to color all vertices red or blue so that each edge has endpoints of differing colors. It is guaranteed that such coloring strategy always exists, i.e., the graph is bipartite. It is a simple task indeed, but Walk Alone colored some vertices incorrectly, so he is asking you for help.

In his bipartite graph, some vertices are colored red, some are colored blue, and the others are not colored yet. Since a colored vertex cannot change its color, your task is to remove as few vertices as possible (as well as the edges connecting with them), so that it is possible to color all uncolored vertices, and each edge should have endpoints of differing colors after that.

### Input

The first line contains two integers  $n$  ( $1 \leq n \leq 10^4$ ) and  $m$  ( $1 \leq m \leq 10^4$ ), the number of vertices and edges in the original graph.

The second line contains  $n$  integers. The  $i$ -th integer  $c_i$  indicates the color of the  $i$ -th vertex: 1 for red, 2 for blue, and 0 if the vertex is not colored.

Each of the next  $m$  lines contains two integers  $u$  and  $v$  ( $1 \leq u, v \leq n$ ), indicating an edge between vertex  $u$  and  $v$ . It is guaranteed that the given graph is bipartite.

### Output

Print the minimum number of vertices you need to remove in the first line, and the label of removed vertices separated by spaces in the second line. If there exists multiple solutions, output any of them.

### Examples

| standard input                                                        | standard output |
|-----------------------------------------------------------------------|-----------------|
| 7 7<br>1 2 1 0 2 1 0<br>1 5<br>2 5<br>3 5<br>3 6<br>4 6<br>2 7<br>3 7 | 2<br>3 5        |
| 5 4<br>1 1 2 2 0<br>5 1<br>5 2<br>5 3<br>5 4                          | 1<br>5          |
| 3 2<br>0 0 1<br>1 2<br>2 3                                            | 0               |

## Note

Here is an illustration for the first example:

