CS102A Spring 2020 Assignment 2

Problem 1. Alice's Gift [Medium, 20 marks]

Description

Alice wants to buy gifts to her parents.

There are n gifts in the shop, the prices of which are $a_1, a_2, \dots, a_n$, respectively. Alice has m yuan, and she wants to buy **two** gifts. In order not to waste money, Alice wants to choose two gifts so that she can spend exactly m yuan to get the gifts. However, she is not good at math and wonders if you can help her.

Input

Using `System.in`

The first line contains two integers n, m.

The second line contains n integers denoting $a_1, \dots, a_n$.

It is **guaranteed** that $0 < n \le 5000, 0 < m \le 2 \times 10^7, 0 < a_1 < a_2 < \cdots < a_n \le 10^7$, and there exists a **unique** answer.

Output

Two integers i, j $(1 \le i < j \le n)$, separated by a space, denoting $a_i + a_j = m$.

Sample Input

```
6 9
1 2 3 5 7 9
```

Sample Output

```
2 5
```

Explanation

$a_2 + a_5 = 2 + 7 = 9$

Brute force solution is acceptable in this problem. For those *who have programming experiences*, **binary search** algorithm is suggested.

Problem 2. B.U.G. [Medium, 20 marks]

Description

B.U.G. is short for “Bug Universal Generator”.

Bob is a professional hacker, and he wants to hack into a buggy online judge system to fetch the test data. He has found out that the system is using *B.U.G.* to generate the serial ID: The generator holds a serial of **secret keys** $(s_1, k_1), (s_2, k_2), \dots, (s_n, k_n), \dots$ and an **initial seed** a_0. Then the serial ID is generated using the formula:

$$a_i = s_i a_{i-1} + k_i$$

where $i > 0$. Since the ID will be too long to store, the system also holds a **modular seed** m, so the actual ID is:

$$a_i = (s_i a_{i-1} + k_i) \bmod m$$

where $i > 0$.

Bob has managed to get the stream of data containing the values mentioned above, but he thinks that calculating the ID is way too easy, so he hires you to do the simple task for him. Of course you will never have access to the test data, but you can get a high mark in this assignment.

Input

Using `System.in`

The input consists of $n + 2$ lines (the value of n is **not given** in the input).

The first line contains two positive integers a_0, m.

Then the i-th line contains two positive integers s_{i-1}, k_{i-1}, where $i = 2, 3, \dots, n + 1$.

The last line contains a single -1.

It is **guaranteed** that $n < 10^6$, and all the values mentioned above are in the range $[1, 10^6]$.

Hint: Consider using `long` to prevent overflowing.

Output

One integer, representing $a_0 \oplus a_1 \cdots \oplus a_n$ ($\oplus$ represents the binary `xor` operation).

Sample Input

```
4 13
13 3
10 19
10 6
2 1
-1
```

Sample Output

```
10
```

Explanation

For the sample, $n = 4$, and $(a_0, a_1, a_2, a_3, a_4) = (4, 3, 10, 2, 5)$.

So $a_0 \oplus a_1 \oplus a_2 \oplus a_3 \oplus a_4 = 10$.

Problem 3. Counting Numbers [Easy, 15 marks]

Description

Cinderella wants to go to the ball with her sisters.

However, the wicked stepmother gave her a long string of numbers and told her: “You must count the exact number of all the digits within two hours, or you’ll never be able to go to the ball!”

Cinderella is now desperate. Can you help her?

Input

Using `System.in`

The only line consists of a **non-empty** string s containing digits from 0 to 9.

It is **guaranteed** that the length of s doesn't exceed 10^7.

Output

10 integers $a_0, a_1, \ldots, a_9$, separated by a space, denoting the number of $0, 1, \ldots, 9$ occurred in the string, respectively.

Hint: You may use `toCharArray()` method to obtain the `char` array from `String` for faster access.

Sample Input

```
012345678910
```

Sample Output

```
2 2 1 1 1 1 1 1 1 1
```

Explanation

0 and 1 both appeared twice, the rest of the digits only appeared once.

Problem 4. Duet [Medium, 20 marks]

Description

David is fascinated with sequences. His new book *sequeNce disCOVeries* has just been published.

In this book, he introduced a new concept: In the sequence $\{a_n\}$, if $a_i, a_j (i < j)$ satisfies $a_i \geq a_j$, then the pair (a_i, a_j) is called a **duet**. The product of a_i and a_j is called the **duet value**.

David is an arrogant guy, and he wants to test if you have read the book: He will provide a finite sequence $a_1, \ldots, a_n$, and you should answer the sum of all duet values. If you fail to give the correct answer, he will throw his fist into your face!

Input

Using `System.in`

The first line contains a positive integer n.

The second line contains n positive integers $a_1, \ldots, a_n$.

It is **guaranteed** that $0 < n \leq 2 \times 10^3$, and $0 < a_i \leq 5 \times 10^5$ $(1 \leq i \leq n)$.

Hint: Consider using `long` to prevent overflowing.

Output

One positive integer, denoting the sum of all duet values.

Sample Input

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

Sample Output

```
184
```

Explanation

For the sample, the duets are: $(3,1),(3,2),(5,1),(5,2),(5,4),(6,4),(8,7),(8,4),(7,4)$

So the answer is:
$3\times 1+3\times 2+5\times 1+5\times 2+5\times 4+6\times 4+8\times 7+8\times 4+7\times 4=184$

Brute force solution is acceptable in this problem. For those *who have programming experiences*, **merge sort** algorithm is suggested.

Problem 5. Easy Multiplication [Hard, 25 marks]

Description

Emma likes playing with building blocks - made of matrices!

She plans to build a tower with a couple of matrices. To achieve this, she simply multiplies the matrices one-by-one. Since the process is quite time-consuming, can you indicate the result based on the blocks she is holding?

> If you know how to multiply matrices, you can skip this part.
>
> Suppose we have an $m\times n$ matrix $A=(a_{ij})_{m\times n}$ and an $n\times p$ matrix $B=(b_{jk})_{n\times p}$ (note that the number of rows in B should be equal to the number of columns in A), then we can do the multiplication $C=A\cdot B$, where C is an $m\times p$ matrix and the corresponding entries are:
>
> $$c_{ik}=\sum_{j=1}^{n}a_{ij}b_{jk}\quad(1\le i\le m,1\le k\le p)$$

Input

Using `System.in`

The first line contains an integer N, denoting the number of matrices. Then the following lines describes the n matrices $M_1,M_2,\ldots,M_N$ in order.

For the i-th matrix M_i, the first line contains two positive integers m_i,n_i denoting that the matrix M_i is $m_i\times n_i$. Then for the next m_i lines, each line contains n_i integers, which denotes the entries of the matrix.

For 20% of the data, it is **guaranteed** that $N=2$.

For 100% of the data, it is **guaranteed** that $2\le N\le 6,\ 1\le m_i,n_i\le 60\ (1\le i\le N)$, and $n_{i-1}=m_i\ (2\le i\le N)$. The result is **guaranteed** not exceeding the range of `int` type.

Output

An $m_1 \times n_N$ matrix, representing the total product $\prod_{i=1}^{N} M_i$. Entries in the same row are separated by a space. No other symbols.

Sample Input

```
2
3 2
1 2
2 1
0 -1
2 3
1 0 1
2 1 -2
```

Sample Output

```
5 2 -3
4 1 0
-2 -1 2
```

Explanation

$$\begin{pmatrix} 1 & 2 \\ 2 & 1 \\ 0 & -1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 1 \\ 2 & 1 & -2 \end{pmatrix} = \begin{pmatrix} 5 & 2 & -3 \\ 4 & 1 & 0 \\ -2 & -1 & 2 \end{pmatrix}$$

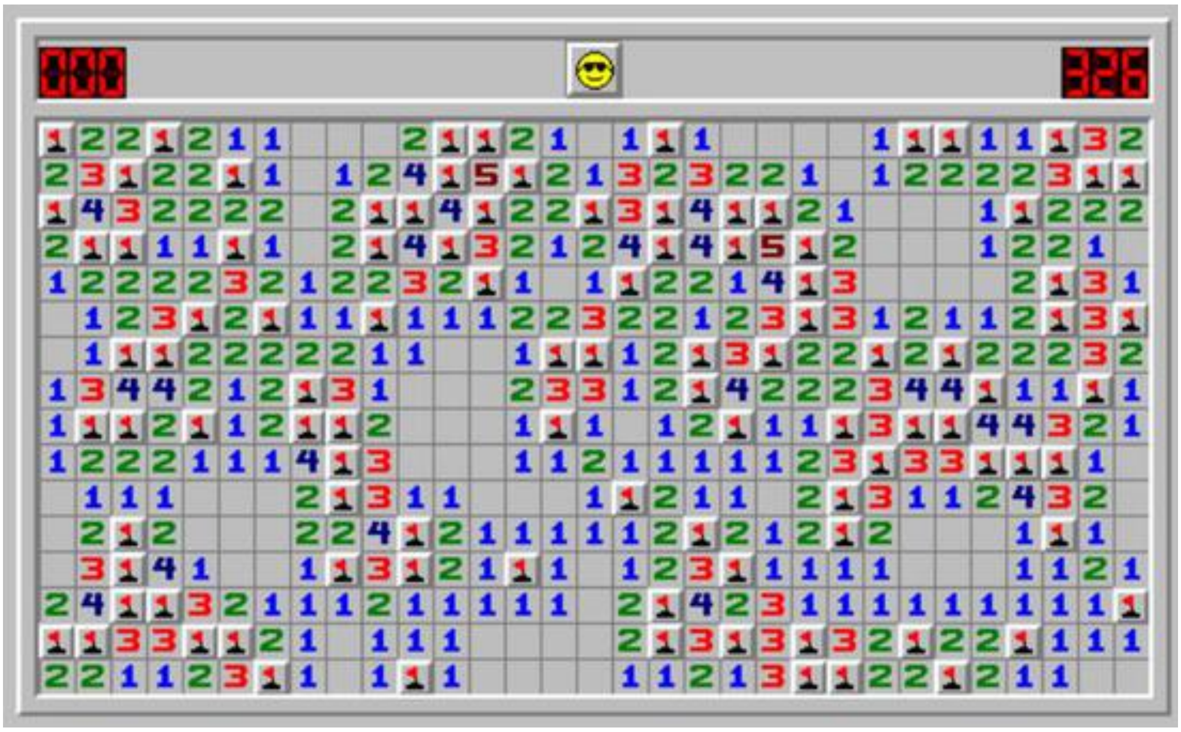
Note that $\prod_{i=1}^{N} M_i = M_1 \cdot M_2 \cdot \dots \cdot M_N$. Matrix multiplication is **not commutative** in most cases.

Problem X. Flags [Bonus, 20 marks]

Description

Have you played *Minesweeper* (扫雷)? It is a game to find out all the bombs on a 2-dimensional map and place a flag on each bomb. The number shown on a block indicates the number of bombs beside the block (i.e. 8 places that are adjacent to the block).

Now given a map with bombs, can you show the map when the game is finished?



Input

Using `System.in`

The first line contains two integers m, n, denoting the size of the map.

For the following m lines, each line contains a string with n characters, where `.` (dot) denotes a blank space and `*` (asterisk) denotes a bomb.

It is **guaranteed** that $10 \leq m \times n \leq 2 \times 10^5$.

Hint: You may use `toCharArray()` method to obtain the `char` array from `String` for faster access.

Output

An $m \times n$ map indicating the result.

Use `F` (capital letter) to denote a flag, numbers `1` to `8` to denote the number of bombs around the block, and `-` (hyphen) to denote blocks with no bombs surrounding.

Sample Input

```
16 31
*..*.......**....*......**..*..
..*..*.....*.*...............**
*........**.*..*.*.**......*...
.**..*...*.*.....*.*.*.........
............*...*....*......*..
....*.*..*...........*......*.*
..**..........**..*.*..*.*.....
.......*..........*.......*..*.
```

```
.**.*..**.....*....*..*.**.....
........*..............*..***..
........*.......*.....*........
..*.......*.......*...*.....*..
..*.....*.*..*.....*...........
..**.............*............*
**..**...........*.*.*..*..*...
......*...*..........**..*.....
```

Sample Output

```
F22F211---2FF21-1F1----1FF11F32
23F22F1-124F5F21323221-122223FF
F432222-2FF4F22F3F4FF21---1F222
2FF11F1-2F4F32124F4F5F2---1221-
122223212232F1-1F2214F3----2F31
-123F2F11F11122322123F312112F3F
-1FF2222211--1FF12F3F22F2F22232
1344212F31---23312F4222344F11F1
1FF2F12FF2---1F1-12F11F3FF44321
12221114F3---1121111123F33FFF1-
-111---2F311---1F211-2F3112432-
-2F2---224F2111112F212F2---1F1-
-3F41--1F3F21F1-123F1111---1121
24FF32111211111-2F4231111111111F
FF33FF21-111----2F3F3F32F22F111
221123F1-1F1----11213FF22F211--
```

Explanation

See the picture shown in the description.