

Problem A: Reborn and HearthStone

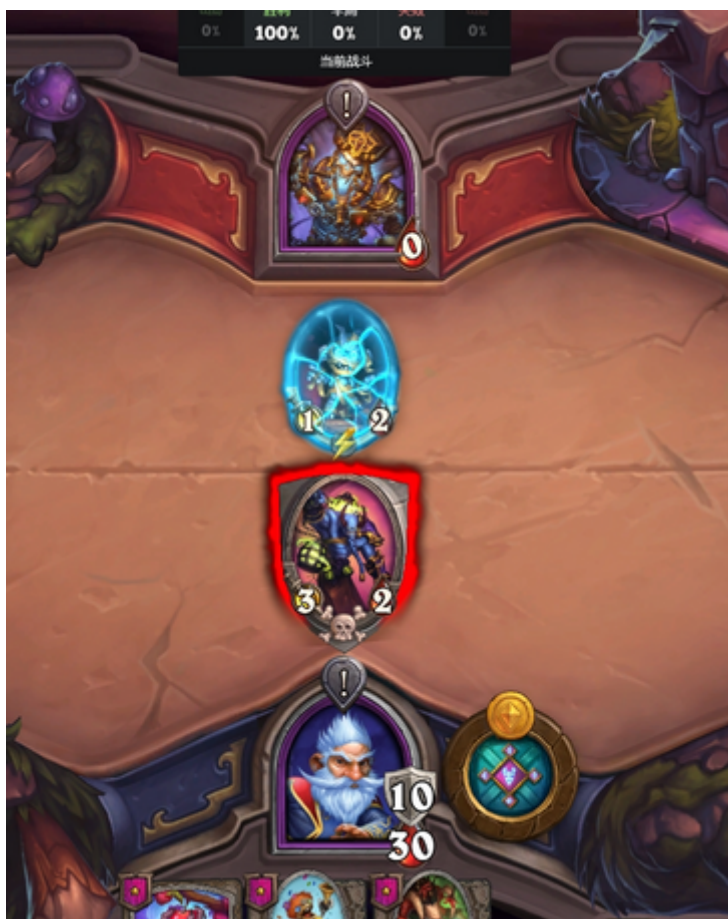
Time limit: 1 second

Memory limit: 1024 megabytes

Reborn is playing Hearthstone Battlegrounds again while slacking off at work. Although there is only one of her minions and one enemy minion on the battlefield, she is still overloaded with decision-making. Given the stats of both minions, determine whether Reborn can win.

- Reborn's minion initially has HP_r health and ATK_r attack points.
- The enemy minion initially has HP_x health and ATK_x attack points.

Each round, Reborn's minion attacks first, reducing enemy's health by ATK_r , while simultaneously losing ATK_x health points itself. The battle ends when at least one minion's health drops to zero or below. Reborn wins if her minion's health remains above zero at that time.



Input

- First line: two integers HP_r, ATK_r .
- Second line: two integers HP_x, ATK_x .

Constraints

- $1 \leq HP_r, ATK_r, HP_x, ATK_x \leq 10000$

Output

Print `Yes` if Reborn wins; otherwise, print `No`.

Example

input

```
1 | 3 2
2 | 1 2
```

output

```
1 | Yes
```

input

```
1 | 114 514
2 | 19 1919
```

output

```
1 | No
```

Problem B: Hile and Fx

Time limit: 2 seconds

Memory limit: 1024 megabytes

Hile throws you a function $f(x)$ defined as the sum of digits of integer x . For example, $f(5) = 5$, $f(14) = 1 + 4 = 5$.

Given an integer y , find a non-negative integer x satisfying: $x + f(x) = y$

If no such x exists, output `-1`.

Input

- First line: integer T , the number of test cases.
- Next T lines: an integer y per test case.

Constraints

- $1 \leq T \leq 2 \times 10^5, 0 \leq y \leq 10^9$

Output

Output a non-negative integer x satisfying $x + f(x) = y$, or `-1` if no such integer exists. If multiple answers exist, output any.

Example

input

```
1 | 6
2 | 11
3 | 45
4 | 14
5 | 19
6 | 19810
7 | 1
```

output

```
1 | 10
2 | 36
3 | 7
4 | 14
5 | 19778
6 | -1
```

Problem C: Reborn and SegmentTree

Time limit: 1.5 seconds

Memory limit: 1024 megabytes

Reborn wrote code involving RMQ (Range Min/Max Query) using a segment tree but encountered TLE (Time Limit Exceeded) . She wants to know how many recursive calls are made by her code for each query.

The segment tree implementation:

```
1  #include <bits/stdc++.h>
2
3  using namespace std;
4  const int N = 2e5 + 10, inf = 0x3f3f3f3f;
5
6  #define min(a, b) ((a) < (b) ? (a) : (b))
7  int tot = 0;
8  int a[N];
9
10 struct SegmentTree{
11     int f[N << 2];
12     void build(int p, int l, int r){
13         if(l == r){
14             f[p] = a[l];
15             return;
16         }
17         int mid = (l + r) >> 1;
18         build(p << 1, l, mid);
19         build(p << 1 | 1, mid + 1, r);
20         f[p] = min(f[p << 1], f[p << 1 | 1]);
21     }
22     int query(int p, int l, int r, int ql, int qr){
23         tot += 1;
24         if(l > qr || r < ql) return inf;
25         if(l >= ql && r <= qr){
26             return f[p];
27         }
28         int mid = (l + r) >> 1;
29         return min(query(p << 1, l, mid, ql, qr), query(p << 1 | 1, mid + 1,
30 r, ql, qr));
31     }
32 }seg;
33
34 int main(){
35     int n, q;
36     cin >> n >> q;
37     for(int i = 1; i <= n; i++){
38         cin >> a[i];
39     }
40     seg.build(1, 1, n);
41     while(q > 0){
42         int l, r;
43         tot = 0;
44         cin >> l >> r;
```

```

45     seg.query(1, 1, n, l, r);
46     cout << tot << endl;
47     q -= 1;
48 }
49 return 0;
50 }
```

Given a sequence and multiple queries, output the value of `tot` for each query.

Input

- First line: two integers n, q .
- Second line: n integers $a_1, a_2, \dots, a_n$.
- Next q lines: two integers l, r per query.

Constraints

- $1 \leq n, q \leq 2 \times 10^5, 1 \leq a_i \leq 10^9, 1 \leq l \leq r \leq n$

Output

For each query, output the value of `tot` made during the query.

Example

input

```

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

output

```

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

Hint

Be careful when using macro definitions.

Problem D: Piza Removes the Letters

Time limit: 1.5 seconds

Memory limit: 1024 megabytes

Piza gives you a lowercase string S . First, you must remove a substring of length m , then remove k additional characters from the remaining string. Your goal is to minimize:

$$\sum_{i=0}^{25} (\text{cnt}(S, 'a' + i))^2$$

where $\text{cnt}(S, x)$ denotes the number of occurrences of character x in string S .

If a string T can be obtained by deleting some letters (possibly zero) from the front and back of S , we say T is a substring of S .

Input:

First line: integer T , number of test cases.

Each test case consists of:

- First line: three integers n, m, k , representing the length of string S , the length of the substring to remove, and the number of additional letters to remove.
- Second line: a string S consisting of lowercase letters.

Constraints:

$$1 \leq T \leq 2 \times 10^5$$

$$1 \leq n \leq 2 \times 10^5, 0 \leq m \leq n, 0 \leq k \leq n - m$$

$$\sum n \leq 5 \times 10^5$$

Output:

Output the minimum value of $\sum_{i=0}^{25} (\text{cnt}(S, 'a' + i))^2$

Example

input

```
1 | 1
2 | 11 3 4
3 | abcaabcbbaa
```

output

```
1 | 6
```

Problem E: Piza Removes Nothing

Time limit: 3 seconds

Memory limit: 1024 megabytes

Piza gives you a tree (an undirected connected graph with n nodes and $n - 1$ edges). Each edge has a weight (possibly negative). Determine how many shortest paths between pairs of nodes have a positive total weight.

Input:

First line: integer n , the number of nodes.

Next $n - 1$ lines: three integers u, v, w , indicating an edge between nodes u and v with weight w .

Constraints:

$$2 \leq n \leq 2 \times 10^5, 1 \leq u, v \leq n, u \neq v, |w| \leq 10^9$$

Output:

Output a single integer representing the number of shortest paths with positive total weight.

Example

input

```
1 | 5
2 | 1 2 -3
3 | 2 3 4
4 | 2 4 1
5 | 4 5 6
```

output

```
1 | 8
```

Problem F: Reborn and TFT

Time limit: 3 seconds

Memory limit: 1024 megabytes

Reborn is obsessed with Teamfight Tactics (TFT).

She currently has n slots, and there are m champions in the card pool. Each champion has several traits among k total traits. To activate a trait, she needs at least x_i distinct champions with that trait on the field.

Given the number of champions she can deploy is at most n , determine the maximum number of traits Reborn can activate.

Input:

First line: three integers n, m, k .

Second line: k integers $x_1, x_2, \dots, x_k$ where x_i is the number of champions required to activate the i^{th} trait.

Next m lines: each line contains a binary string s of length k , representing traits of a champion ($s_j = 1$ if the champion has the j^{th} trait, $s_j = 0$ if the champion doesn't have the j^{th} trait).

Constraints:

$$1 \leq n, m \leq 28, 1 \leq k \leq 14$$

$$1 \leq x_i \leq 2, |s| = k, s_i \in \{0, 1\}$$

Output:

Output a single integer, the maximum number of traits that can be activated.

Example

input

```
1 1 2 4
2 1 1 2 2
3 0011
4 1100
```

output

```
1 2
```

Problem G: Geos Likes Shopping

Time limit: 1.5 seconds

Memory limit: 1024 megabytes

There are n types of items, with a_i available items for type i . You need to take exactly m items. The profit for each item type depends on how many items of this type are taken. If b_i items of type i are taken, each item provides profit $w_i - b_i$, and the total profit is:

$$\sum_{i=1}^n b_i \times (w_i - b_i)$$

Find the maximum possible profit. (Guaranteed $w_i \geq a_i$, and $m \leq \sum a_i$).

Input:

First line: two integers n, m , the number of item types and the number of items to take.

Second line: n integers $w_1, w_2, \dots, w_n$.

Third line: n integers $a_1, a_2, \dots, a_n$, where a_i is the quantity of type i available.

Constraints:

$$1 \leq n \leq 10^5, 1 \leq m \leq 10^9$$

$$1 \leq w_i \leq 10^9, 1 \leq a_i \leq w_i$$

It's guaranteed that $m \leq \sum a_i$.

Output:

Output the maximum possible profit.

Example

input

1	5	6			
2	1	2	3	4	5
3	1	2	3	4	5

output

1	13
---	----

Problem H: Kyooma Loves Tree

Time limit: 1.5 seconds

Memory limit: 1024 megabytes

Kyooma has a perfect k -ary tree of depth n . Each edge has a probability p of disappearing. He wants to calculate the expected number of trees remaining in the forest.

A perfect k -ary tree is defined as each node (except leaf node) having exactly k child node, and all leaf node are at the same depth.

Input:

First line: integer T , number of test cases.

Each of the next T lines contains three integers n, k, p , representing the tree depth, number of child node per node, and the probability of disappearing modulo 998244353.

Constraints:

$$1 \leq T \leq 10^5, 1 \leq n, k \leq 10^{18}, 0 \leq p < 998244353.$$

Output:

Output one line per test case: the expected number of trees modulo 998244353.

Example

input

```
1 | 2
2 | 2 2 499122177
3 | 114514 1919810 0
```

output

```
1 | 2
2 | 1
```

Problem I: Hile and Array

Time limit: 1 second

Memory limit: 1024 megabytes

Given an operation sequence of length n . The i^{th} position contains an operation (addition, subtraction, multiplication) represented by a pair (o_i, p_i) . You are given q queries. Each query provides three integers (x, l, r) . Starting with an initial value x , sequentially perform the operations from position l to r , and calculate the result modulo $10^9 + 7$.

Note: Operations must be executed strictly in the provided order, meaning addition/subtraction earlier in the sequence takes precedence over later multiplications.

- If $o_i = 1$, perform addition with p_i .
- If $o_i = 2$, perform subtraction with p_i .
- If $o_i = 3$, perform multiplication by p_i .

Input

The first line contains an integer T , the number of test cases.

For each test case:

- First line: integer n , the length of the operation sequence.
- Next n lines: each containing two integers (o_i, p_i) , representing operation type and operand.
- Next line: integer q , the number of queries.
- Next q lines: three integers x, l, r .

Constraints:

$1 \leq T \leq 10^4, 1 \leq n, q \leq 10^5, 1 \leq o_i \leq 3, 1 \leq p_i \leq 10^9$
 $1 \leq x \leq 10^9, 1 \leq l \leq r \leq n, \sum n \leq 10^5, \sum q \leq 10^5$.

Output:

For each query, output the result modulo $10^9 + 7$.

Example

input

```
1 | 1
2 | 6
3 | 1 1
4 | 2 1
5 | 3 4
6 | 1 5
7 | 2 1
8 | 3 4
9 | 2
10 | 1 1 6
11 | 100 3 4
```

output

1	32
2	405

Problem J: Boboge and Card Shuffle

Time limit: 1 second

Memory limit: 1024 megabytes

Boboge has a deck of cards consisting of four suits (H, Z, C, U). Each suit has exactly n cards numbered from 1 to n , totaling $4n$ cards. He wants to shuffle the cards into a new arrangement satisfying two conditions:

- The subsequence of each suit must have increasing numbers.
- The last n cards must all be of suit 'U'.

In each operation, Boboge can take one card and insert it anywhere in the deck.

Determine the minimum number of operations required.

Input:

First line: integer T , the number of test cases.

For each test case:

- First line: integer n .
- Second line: $4n$ strings representing the cards. Each string consists of a suit letter (H, Z, C, U) followed by its number (e.g. H1).

Constraints:

$$1 \leq T \leq 10^4, 1 \leq n \leq 10^5, \sum n \leq 10^5.$$

Output:

For each test case, output the minimum number of operations required.

Example

input

```
1 3
2 2
3 H1 Z1 C1 C2 Z2 H2 U1 U2
4 2
5 H1 Z1 C1 U1 H2 Z2 C2 U2
6 10
7 C10 C2 Z3 H7 C8 Z7 U1 C3 C5 H10 U2 U4 Z6 H9 Z4 Z2 U3 H5 U8 Z5 H8 C7 C6 C4 Z9
   U6 C1 H4 U5 Z8 H3 U9 H2 H1 U10 H6 Z1 Z10 C9 U7
```

output

```
1 0
2 1
3 27
```

Hint

The same color (excluding **U**) does not necessarily have to be arranged in adjacent positions, for example, **H1 Z1 C1 H2 C2 Z2 U1 U2** is a legal sorting method.

Problem K: Hile counts the Qi

Time limit: 3 seconds

Memory limit: 1024 megabytes

In the game of Go, "atari" refers to placing a stone so that exactly one "Qi" remains for the connected group of stones adjacent to it. The number of "Qi" refers to the number of empty spaces in the up, down, left, right directions of a stone that do not yet have any chess pieces, and is not considered Qi outside the board. A group of stones shares Qi. A "group" refers to connected stones of the same color (four-connected adjacency).

Given a Go board with black (**B**) and white (**W**) stones (and empty spaces marked by **E**), calculate the number of groups in "atari" for both black and white stones separately.

Input:

First line: integer n , the size of the board ($n \times n$).

- Next n lines: each line contains n characters (**E**: empty, **W**: white stone, **B**: black stone).

Constraints:

$$2 \leq n \leq 1000$$

Output:

Output two integers representing the number of black and white groups in "atari".

Example

input

```
1 5
2 BWEEW
3 BBWE
4 WEEBW
5 BBWEE
6 WBWBE
```

output

```
1 1 2
```

input

```
1 5
2 WBEEW
3 EWBBB
4 BEWBE
5 EEWBB
6 BBEBE
```

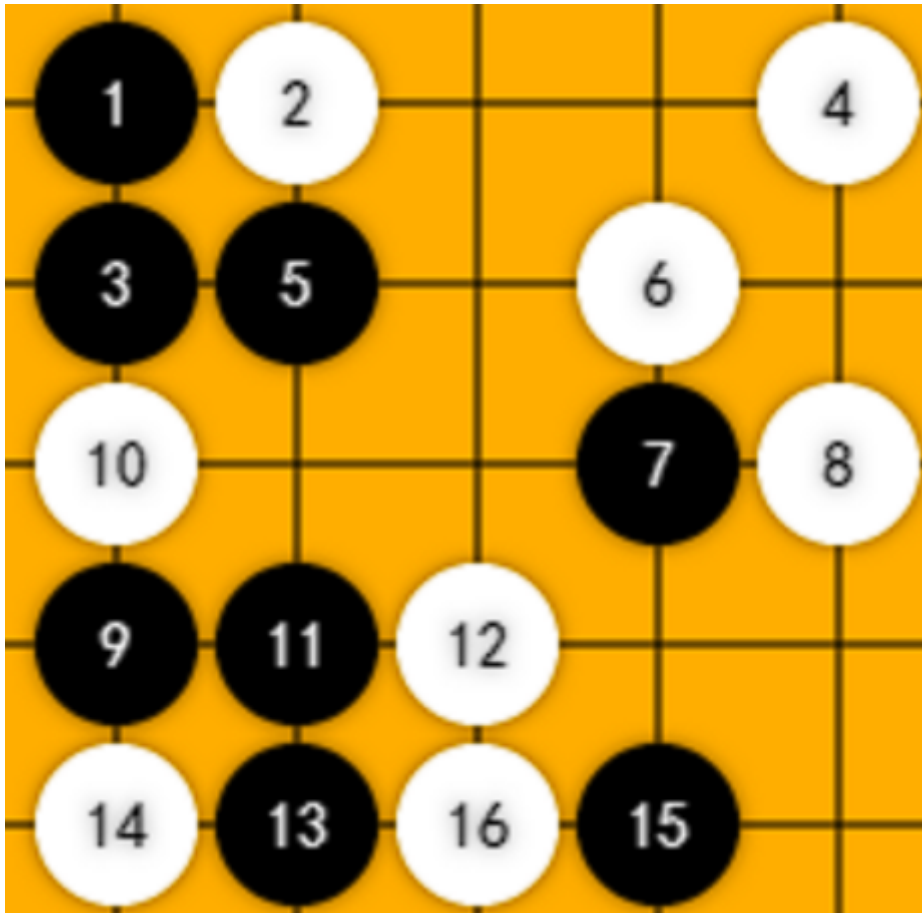
output

1 | 0 4

Hint

In Example 1, if the black stone of the first move located in the upper left corner is taken as the coordinate (1, 1), and the black stone of the third move is taken as the coordinate (2, 1), then the "atari" of the black stone is located at (1, 4); The "atari" of white is located at (3, 2) and (5, 5).

The three stones located at (1, 1), (2, 1), and (2, 2) are one group of stones, and they all have the two "Qi" of (2, 3) and (3, 2).



Problem L: Kyooma Loves Numbers

Time limit: 1 second

Memory limit: 1024 megabytes

Kyooma likes numbers divisible by 9. You are given several numbers represented in base-15. You may perform at most one operation: choose two different positions in the number and swap their digits.

For example, given a base-15 number `C08`, you can obtain: `C08`, `C80`, `0C8`, or `80C`.

Determine if it is possible, using at most one such operation (possibly none), to make the number divisible by 9 (decimal).

Input:

The first line contains a positive integer T , representing the number of test cases.

- Each test case consists of one line containing a base-15 integer n .

Constraints:

- $1 \leq T \leq 20$
- $0 \leq n \leq 15^{500^2}$

Digits A, B, C, D, E represent 10, 11, 12, 13, 14 respectively.

Output:

For each test case, output `YES` or `NO`.

Example

input

```
1 | 6
2 | 9
3 | DD
4 | 109
5 | 36
6 | 53
7 | 17
```

output

```
1 | YES
2 | NO
3 | YES
4 | NO
5 | NO
6 | NO
```

Problem M: Kyooma Loves Numbers II

Time limit: 1 second

Memory limit: 1024 megabytes

Given an integer n , provide five positive integers a_1, a_2, a_3, a_4, a_5 , each no greater than 10^{18} , satisfying:

$$\sum_{i=1}^5 \frac{1}{a_i} = \frac{4}{n}, a_i \neq a_{i+1} \quad (1 \leq i \leq 4), \quad a_2 \neq a_4$$

If multiple solutions exist, output any one. If no solution exists, output `-1`.

Input:

First line: integer T , the number of test cases.

Next T lines: each contains an integer n .

Constraints:

- $1 \leq T \leq 5000$
- $1 < n \leq 10^7$

Output:

Output five integers satisfying the conditions above, or `-1` if no solution exists.

Example

input

```
1 | 2
2 | 3
3 | 5
```

output

```
1 | 2 3 7 42 3
2 | 5 16 3 5 240
```