CS F211
Data Structures and Algorithms
Assignment - 7

Allowed languages: **C**

March 12, 2021

General Tips

- Try to use functions as much as possible in your code. Functions increase reusability and the pass-by-value feature provides a significant help sometimes. Modularizing your code also helps you to debug efficiently.

- Use `scanf` to read characters/strings from STDIN. Avoid using `getchar`, `getc` or `gets`. Try to read up about character suppression in `scanf` as it will be very helpful in some of the problems.

- Use `printf` instead of `putc`, `putchar` or `puts` to print character/string output on STDOUT.

- Indent your code appropriately and use proper variable names. These increase readability and writability of the code. Also, use comments wherever necessary.

- Use a proper IDEs like Sublime Text or VSCode as they help to run and test your code on multiple test-cases easily. You can install Windows Subsystem Linux (WSL) or MinGW 7.3.0, if you are Windows user to compile and run your programs. Alternatively, you can run and test your codes on Online GDB. If you are using WSL or Linux to run your programs, make sure that the gcc version is `gcc 5.4.1 c99`.

A: Another Amulet

Nathaniel has given Bartimaues another arduous task that involves breaking into the house of Simon Lovelace. He can't transform to sneak in otherwise Faquarl will notice, so the only way to enter is to break the security code to his house. The security code gives you N numbers in the form of an array and to unlock it you must do the following: For each element in the array, output the index of the closest element on the left that is smaller than it. If no such element exists (that is, there is no element smaller than the current element to the left of the current element) then output -1. Note that the output must be 1-indexed. *Hint: See if you can use a stack to solve this in linear time.*

Input

The first line contains a single integer N as described above $(1 \le N \le 10^5)$. The second line of input contains N integers where the ith number represents a_i $(1 \le a_i \le 10^9)$.

Output

Print N space separated integers as described above.

---

input
7
2 5 3 7 8 1 9

output
-1 1 1 3 4 -1 6

explanation
for the first element, there is nothing to the left smaller than it, so the answer
for it is −1. For the second element, the first element that is smaller than it is
in the first position, so the answer is 1. For the third element, the first element
that is smaller than it to the left is also at position 1 so the answer for that is
also 1 and so on...

B: The Eye

There's another Golem loose in London and Nathaniel has to take care of it since he's the head of Internal Affairs. Upon arriving at the scene, he sees that the Golem has smashed through a row of shops simply by breaking through the walls. Peculiarly, the heights of the holes in the walls are not the same. Suppose that there are N walls in a row and the distance between each wall is 1 unit. Also assume that the width of each hole is 1 unit. The height of the hole in the ith wall is a_i. To estimate the dimensions of the Golem (which is more or less cuboidal in shape), Nathaniel needs to find the volume of the largest possible cuboid that can fit in this entire setup. More formally, find the volume of the largest cuboid (by volume) that can fit in any contiguous segment of these holes without exceeding the dimensions of the smallest hole in this subsegment. The holes are rectangular and all holes start from the ground level. *Hint: Try using the idea from the previous problem here*

Input

The first line contains an integer N representing the number of holes ($1 \leq N \leq 10^5$). The next line contains N numbers representing the height of the ith hole h_i ($1 \leq a_i \leq 10^9$).

Output

Print a single integer, the volume of the largest possible cuboid (by volume) that can fit in the given set of holes

---

```
input
7
6 2 5 4 5 1 6

output
12
```

```
explanation
Here it is best to fit a cuboid of length 3 and height 4 on the indices 3, 4, 5 (1-indexed).
We can clearly see that it fits in this subsegment and it's total volume is 12.  It
is not possible to get a higher volume in this test case.
```

C: Nouda's Minions

Now that Nathaniel and Bartimaeus have joined forces (literally), they have to work together to eradicate all the demons running around St. James Park in London. The park can be represented by a grid where each cell of the grid is occupied by either a 1 or a 0. 1 represents a demon in that cell and 0 represents a human. They have one burst of energy left in the Staff of Gladstone. A shot from this staff will eliminate all living things within some rectangle on the grid (you can choose the rectangle in any way, and a rectangle consists of multiple cells). Bartimaeus and Nathaniel want to kill as many demons as possible with this last shot so they need you to find the rectangle that contains the largest number of 1s and does not contain even a single 0. *Hint: Try using the idea from the previous problem here*

Input

The first input line has two integers N ($1 \leq N \leq 10^3$) and M ($1 \leq M \leq 10^3$): the dimensions of the park The next N lines contain M integers each, all either 0 or 1.

Output

Output a single integer representing the largest number of 1s in a rectangle that does not contain any 0s.

```
input
3 3
1 1 0
1 1 1
1 1 0

output
6


explanation
Here the largest such rectangle is from (1,1),(1,2),(2,1),(2,2),(3,1),(3,2) (1-indexed).
Any larger rectangle would contain 0s which is not allowed.
```

D: Getting Rid of the Ring

Bartimaeus is trying to reach the ocean to throw away the ring before Ammet catches him. The ocean is too far away so he decides to throw it in a nearby lake instead. Let the Land be represented by a grid with 1s and 0s, where a 1 means that there's a lake in that cell and 0 is just plain land. Bartimaeus is a little confused about where he is right now so he asks you to find the nearest manhattan distance to **any** lake on the grid *for every point on the grid*.

Take a look at the following concept to help you out: Multi Source BFS

Input

The first line contains two integers N and M $(1 \leq N, M \leq 10^5)$. The next N line contain M integers each, all the of them being either 1 or 0.

Output

Print N lines of M integers each, the manhattan distance of every cell to the nearest lake. If the cell itself is a lake then the nearest distance is 0.

```
input
4 4
0 0 0 1
0 1 0 0
0 0 1 1
0 0 0 0

output
2 1 1 0
1 0 1 1
2 1 0 0
3 2 1 1

explanation
for the cell (1,1)(1-indexed), the nearest lake is at cell (2,2).  For the cell (1,3)
the nearest lake is at the position (1,4), and so on.
```

E: The Masked Biter

You are initially given an empty multiset of integers (Remember that a multiset is a collection of numbers without any particular ordering that may or may not have duplicate values. There are three types of operations that are performed:

- add x: this inserts the number x into the multiset

- delete x: this removes exactly one occurrence of the element x (it is guaranteed that at least one element exists in the multiset with the value x before performing this operation).

- query x: You have to output the number of elements in the multiset that are **parity equivalent** with x.

To determine whether two numbers are parity equivalent or not, first make the number of digits of both numbers equal by appending zeroes to the left of the smaller number. Once they have an equal number of digits, they are parity equivalent iff the parity (whether the digit is even or odd) of every digit of one number is the same parity as the corresponding digit in the other number.

Input

The first line contains a single integer N representing the number of operations performed ($1 \leq N \leq 10^5$). The next N lines each contain a string representing the type of operation and a number x representing the number used in that operation ($1 \leq x \leq 10^9$).

Output

For every "query" operation, output the answer for that query in a single line.

---

input
6
add 241
query 1
add 361
delete 241
query 2301

output
1
1

explanation
For the first query, the integer in the multiset at that time is 241. Since the value in the query matches it, the answer is 1. For the second query, the only number that matches the query is 361. Hence the answer is 1

F: Another String Game

Alice & Bob are having a friendly competition to see who knows more words. In this turn-based game, Alice begins by saying out loud a word, S. Bob responds by saying another word that begins with the last letter of the previous word. They continue this process in an alternating fashion till one of them is unable to come up with a new word, which is when the game ends. Given a sequence of words (one indexed) that they used in the game, you need to determine the winner. Rules:

- From the 2nd word onwards, each word *must* be unique, i.e. not used before. If anyone uses a word that's already been used before, they lose. Additionally, the $(i+1)^{th}$ word *must* begin with the last letter of the ith word ($\forall i \geq 2$) - otherwise, the person who said the $(i+1)^{th}$ word loses, and further words do not need to be checked.

- Words are one indexed, and the starting word (the first word) S can be any word. If all words are valid, the person who said the last word wins. Find out who wins each game.

Hint: Look up tries.

Input

The first line contains T ($0 \leq T \leq 10$), the number of testcases. Each testcase begins with a line that contains one integer N ($1 \leq N \leq 5 \times 10^5$), the number of words played in total before the game ends. The next line contains N space-separated strings, the list of words used. (odd items in this list correspond to words Alice said, and even items correspond to Bob's words). All strings will be of length not greater than 10 and contain only English lowercase letters.

Output

Output T space separated strings, each of which is the winner of that round - "Alice" or "Bob".

```
input
2
3
apple orange enemy
6
app potato open nose even night
2
yay yay
output
Alice Bob Alice
explanation
Alice wins in the first game since the word "orange", which Bob uses is not a valid
word.  Bob wins in game 2 since he has the last word.  Alice wins in round 3 since
Bob reuses the word "yay".
```

G: Linking Problems

You are given a linked list containing N elements, and a series of k numbers - $a_1, a_2, \ldots, a_k$ - such that $\sum_i a_i = N$. Process this linked list by: reversing the first a_1 elements in the linked list, and multiplying all the elements by a_1. Once this segment is processed, reverse the next a_2 elements and multiply them with a_2, and proceed to next segments in a similar fashion. Output the processed linked list. *Note: You must use a linked list to solve this problem.*

Input

The first line contains N, the number of elements in your linked list, and k, the number of segments $(1 \leq k \leq N \leq 5 \times 10^5)$. The next line contains N space separated values denoting the values in the linked list (all values between -10^3 and $+10^3$). The third line of input contains k numbers, $a_1, a_2, a_3 \ldots, a_k$.

Output

Output one line with N space separated integers, denoting the modified list.

```
input
7 2
1 2 3 -1 -1 4 5
3 4
output
9 6 3 20 16 -4 -4

input
5 5
1 2 3 4 5
1 1 1 1 1
output
1 2 3 4 5
```

H: Engineering Graphics With Trees

You are given a binary tree, represented in the form of an array A, where $A[0]$ represents the root. Additionally, for any node $A[i]$, $A[2i+1]$ and $A[2i+2]$ are the left and right children respectively. All nodes have positive integer values only, and a -1 indicates that no node exists there.

Assume that:

- all nodes at the same *height* from the root are in the same horizontal line.

- an observer looking from the left can only see one node at each height, i.e. the leftmost node. Similarly, an observer on the right side of the binary tree can only see the rightmost node at each height.

Find the *left view* and the *right view* of the binary tree - sum up all the values that can be seen from the right and call this R. Similarly, sum up all the values seen from the left and call this L. Output $abs(L - R)$.

Input

The first line contains an integer N ($1 \leq N \leq 10^5$), the size of array A. The next line contains N space separated integers denoting the value in each of the nodes. Also, $-1 \leq A_i \leq 10^5, \forall i$.
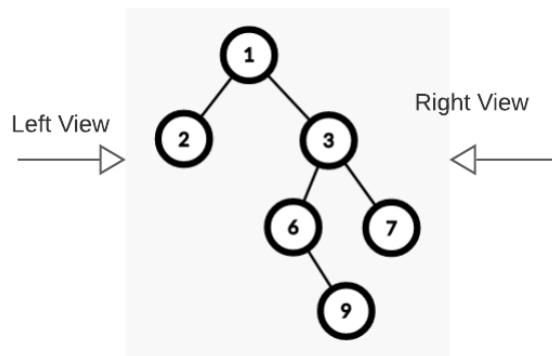
Output

Output one line, consisting of only lowercase English alphabets, denoting the processed string.

```
input
9
1 2 3 -1 -1 6 7 -1 9

output
2
explanation
The nodes that can be seen from the
left are 1, 2, 6, 9.  So, L = 18.
Similarly, R = 20.  abs(L-R) = 2.
```

I. Lazy Day at the Beach

The state of Florida wants to make its beaches safer from crocodiles by posting armed lifeguards at certain places on the beach. The state identified N possible positions along the beach to potentially post guards - $x_1, x_2, x_3, \ldots x_N$ which are given in ascending order (guaranteed that $x_i < x_j$ if $i < j$). Placing a lifeguard at these positions will increase the "safety value" of the beach by $s_1, s_2, s_3, \ldots s_N$ respectively. Two guards cannot be posted within X distance of each other, since they will end up chit-chatting, and ignore the safety of the beach - the distance between any two guards must be **strictly** greater than L. Calculate the maximal safety value possible, assuming you can hire as many guards as you want, provided no two guards you hire are positioned within L distance of each other.

Hint: Let A_k be the answer (maximal safety value possible) if we can use only the first k positions on the beach $(x_1, x_2, \ldots, x_k)$. Clearly, $A_1 = s_1$. Now, let us say we know all values till A_k. To calculate A_{k+1}, we have two options: (1) place this $(k+1)^{th}$ lifeguard, and ignore all lifeguards that were placed from $(x_{k+1} - L)$ to $(x_{k+1} - 1)$ or (2) do not place lifeguard A_{k+1}. Select the maximal of these two options. For example, if $x_{k+1} - x_k > X$, adding the security guard at x_{k+1} will not require removal of any previously placed guards.

Input

The first line of input contains space separated integers N, L. $(1 \leq N \leq 10^4)$. The second line contains N space separated integers, the possible positions $(1 \leq x_i, L \leq 10^5, \forall i)$. The third line contains N space separated integers, the safety values for each of the positions $1 \leq s_i \leq 10^4$.

Output

Print a single integer S denoting the maximal safety value that can be achieved.

---

input
5 5
6 7 12 13 14
5 6 5 3 1
output
10

---

input
4 2
6 9 12 14
5 6 3 7
output
18

J. Merkle Tree

You are given N unsigned positive integers as input in array A. Begin by multiplying the first two integers $A[0]$ and $A[1]$ - then *hash* the product (hash definition given below). Repeat this process for the next two elements, the next two elements and so on. If there is a single integer left over, hash it without multiplying it with anything else. Performing this process on the array A of size N gives us an array of hashes B_1 containing $N/2$ or $\frac{N-1}{2} + 1$ hashes (if N is even or odd respectively). Now, process B_1 in a similar fashion to form B_2. Keep repeating the pairing and hashing process till we have only element (hash) left. Return this single value. The hash definition is given by,

```
def hash(x):
    x = x%65535
    x *= (x + 13);
    x = x%(65535);
    return x
```

Note: The value that you calculated is called the *merkle root or merkle hash*. This data structure, of a hierarchy of hashes, is one of the building blocks of blockchains. Additional reading.

Input

The first line of input contains an integer N ($0 \le N \le 10^3$). The next line contains N space separated integers ($1 \le A_i \le 10^3$)

Output

Print a single integer denoting the calculated single hash left at the end.

---

input
4
123 456 245 90

output
31905
explanation
Initially begin by multiplying $123 \times 456 = 56088$. Hash(56088) = 60933. Similarly, Hash(245×90) = 22845. Now, we have two hashes left – 60933 and 22845. The final answer is Hash($60933 * 22845$) = 31905