Sample-1: Easy

You have an array **A** of integers with **n** elements. There are **q** queries to process and each **query** consists of four integers: **l**, **r**, **x**, and **y**.

For the subarray of **A** ranging from index **l** to **r**, you need to assign a **sequence** of integers for each subsequent element. The **sequence** should start from **x** and increase by **y.** This means:

- **A[l]** will be assigned the value of **x**.
- **A[l+1]** will be assigned the value of **x + y**.
- **A[l+2]** will be assigned the value of **x + 2*y**.
- Continuing this pattern, **A[l+i]** will be assigned the value of **x + i*y**, where **i** ranges from **0 to (r - l)**.

Find the **sum of all integers** in **A** after processing all queries. Since answer can be large, return it **modulo 10^9+7**.

Input Format

1. The first line contains an integer, n, denoting the number of elements in A.
2. Each line i of the n subsequent lines (where 0 ≤ i < n) contains an integer describing A[i].
3. The next line contains an integer, q, denoting the number of rows in queries.
4. Each line i of the q subsequent lines (where 0 ≤ i < q) contains 4 space separated integers each describing the row queries[i].
5. The 4 space separated integers denote the value of l, r, x and y for the i-th query.

Constraints

- $1 <= \boldsymbol{n} <= 10^5$
- $0 <= \boldsymbol{A[i]} <= 10^9$
- $1 \leq \boldsymbol{q} \leq 10^5$
- $0 <= \boldsymbol{queries[i][j]} <= 10^5$

Sample Input 1

```
5
5
5
0
3
0
5
0 2 1 2
0 1 6 5
2 3 8 0
2 4 9 6
3 4 8 9
```

Sample output 1

```
51
```

Explanation-1

Here, `n = 5`

```
A = [5, 5, 0, 3, 0]
q = 5
queries = [[0, 2, 1, 2], [0, 1, 6, 5], [2, 3, 8, 0], [2, 4, 9, 6], [3, 4, 8, 9]]
```

for query 1:

```
  l = 0, r = 2, x = 1, y = 2
  A[0] = 1
  A[1] = 3
  A[2] = 5
```

So, `A = [1, 3, 5, 3, 0]`

for query 2:

```
  l = 0, r = 1, x = 6, y = 5
  A[0] = 6
  A[1] = 11
```

So, `A = [6, 11, 5, 3, 0]`

for query 3:

```
  l = 2, r = 3, x = 8, y = 0
  A[2] = 8
  A[3] = 8
```

So, `A = [6, 11, 8, 8, 0]`

for query 4:

```
l = 2, r = 4, x = 9, y = 6

A = [6, 11, 9, 15, 21]
```

for query 5:

```
l = 3, r = 4, x = 8, y = 9

A = [6, 11, 9, 8, 17]
```

Hence, answer is `6+11+9+8+17 = 51`

Sample Input 2

```
5
3
9
2
5
4
5
1 2 6 3
1 2 2 8
1 2 5 5
1 3 1 8
1 2 2 9
```

Sample output 2

```
37
```

Explanation 2

Here, `n = 5`

```
A = [3, 9, 2, 5, 4]
q = 5
queries = [[1, 2, 6, 3], [1, 2, 2, 8], [1, 2, 5, 5], [1, 3, 1, 8], [1, 2, 2, 9]]
```

for query 1:

```
  l = 1, r = 2, x = 6, y = 3
```

So, A = [3, 6, 9, 5, 4]

for query 2:
l = 1, r = 2, x = 2, y = 8
So, A = [3, 2, 10, 5, 4]

for query 3:
l = 1, r = 2, x = 5, y = 5
So, A = [3, 5, 10, 5, 4]

for query 4:
l = 1, r = 3, x = 1, y = 8
So, A = [3, 1, 9, 17, 4]

for query 5:
l = 1, r = 2, x = 2, y = 9
A = [3, 2, 11, 17, 4]

Hence, answer is 3+2+11+17+4 = 37

Sample Input 3

```
5
0
1
0
0
1
5
1 2 7 7
0 1 3 6
1 1 1 1
3 4 9 1
2 3 1 0
```

Sample output 3

```
16
```

Explanation 3

Here, n = 5
A = [0, 1, 0, 0, 1]
q = 5
queries = [[1, 2, 7, 7], [0, 1, 3, 6], [1, 1, 1, 1], [3, 4, 9, 1], [2, 3, 1, 0]]

for query 1:
l = 1, r = 2, x = 7, y = 7
A = [0, 7, 14, 0, 1]

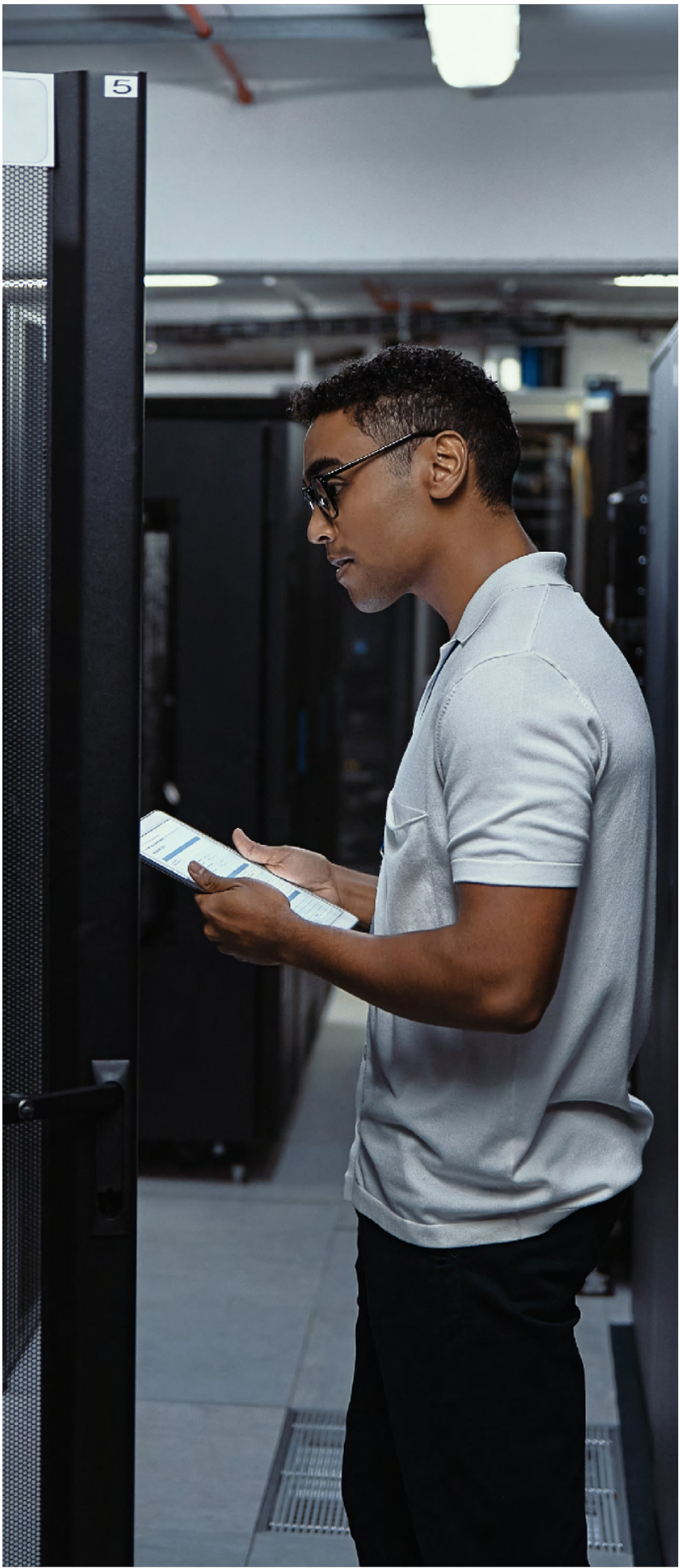
for query 2:
l = 0, r = 1, x = 3, y = 6
A = [3, 9, 14, 0, 1]

for query 3:
l = 1, r = 1, x = 1, y = 1
A = [3, 1, 14, 0, 1]

for query 4:
l = 3, r = 4, x = 9, y = 1
A = [3, 1, 14, 9, 10]

for query 5:
l = 2, r = 3, x = 1, y = 0
A = [3, 1, 1, 1, 10]

Hence, answer is 3+1+1+1+10 = 16



External Document © 2024 Infosys Limited

Sample 2: Medium

You are given three integers **X**,**Y** and **Z** and two arrays **A** and **B** both of length **N**. You are also given an integer **sum** which is initially equal to 0.

You have perform **N operations** and in each **ith** operation you must do **only one** of the following :

1. Subtract **B[i]** from sum.
2. Decrease both of **X** and **Y** by 1, then add **A[i] * X * Y * Z** to **sum**.
3. Decrease both of **Y** and **Z** by 1, then add **A[i] * X * Y * Z** to **sum**.

However, after each operation, **X**,**Y** and **Z** must all remain greater than or equal to 0.

Find the **maximum sum you can obtain after performing all operations**. Since answer can be large, return it **modulo 10^9+7**.

Input Format

1. The first line contains an integer, N, denoting the number of operations.
2. The next line contains an integer, X.
3. The next line contains an integer,Y.
4. The next line contains an integer, Z.
5. Each line i of the N subsequent lines (where 1 ≤ i ≤ N) contains an integer describing A[i].
6. Each line i of the N subsequent lines (where 1 ≤ i ≤ N) contains an integer describing B[i].

Constraints

- $1 <= N <= 10^3$
- $1 <= X <= 10^3$
- $1 <= Y <= 10^3$
- $1 <= Z <= 10^3$
- $1 <= A[i] <= 10^6$
- $1 <= B[i] <= 10^9$

Sample Input-1:

```
2
1
2
2
0
0
10
5
```

Sample output-1:

```
0
```

Explanation-1:

Here, `N = 2, X = 1, Y = 2, Z = 2`

`A = [0, 0]`

`B = [10, 5]`

It is given that in starting, `sum = 0`

operation 1:

Apply type 2 operation (i.e. Decrease both of `X` and `Y` by `1`, then add `A[1]*X*Y*Z` to `sum`)

`X = 0, Y = 1, Z = 2`

`sum = sum + 0*0*1*2 = 0`

operation 2:

Apply type 3 operation (i.e. Decrease both of `Y` and `Z` by `1`, then add `A[2]*X*Y*Z` to `sum`)

`X = 0, Y = 0, Z = 1`

`sum = sum + 0*0*0*1 = 0`

Hence, answer is the final value of `sum` i.e. `sum = 0`.

Sample Input-2:

```
2
10
11
11
1
10
10
0
```

Sample output-2:

```
9990
```

Explanation-2:

Here, `N = 2, X = 10, Y = 11, Z = 11`

`A = [1, 10]`

`B = [10, 0]`

It is given that in starting, `sum = 0`

operation 1:

Apply type 1 operation (i.e. Subtract `B[1]` from `sum`.)

`sum = sum - 10 = -10`

operation 2:

Apply type 3 operation

(i.e. Decrease both of `Y` and `Z` by `1`, then add `A[2]*X*Y*Z` to `sum`)

`X = 10, Y = 10, Z = 10`

`sum = sum + 10*10*10*10 = 9990`

Hence, answer is the final value of `sum` i.e. `sum = 9990.`

Sample Input-3:

3
3
3
3
1
2
3
1
2
3

Sample output-3:

35

Explanation-3:

Here, `N = 3, X = 3, Y = 3, Z = 3`

`A = [1, 2, 3]`

`B = [1, 2, 3]`

It is given that in starting, `sum = 0`

operation 1:

Apply type 1 operation

(i.e. Subtract `B[1]` from `sum`.)

`sum = sum - 1 = -1`

operation 2:

Apply type 2 operation (i.e. Decrease both of `X` and `Y` by `1`, then add `A[2]*X*Y*Z` to `sum`)

`X = 2, Y = 2, Z = 3`

`sum = sum + 2*2*2*3 = 23`

operation 3:

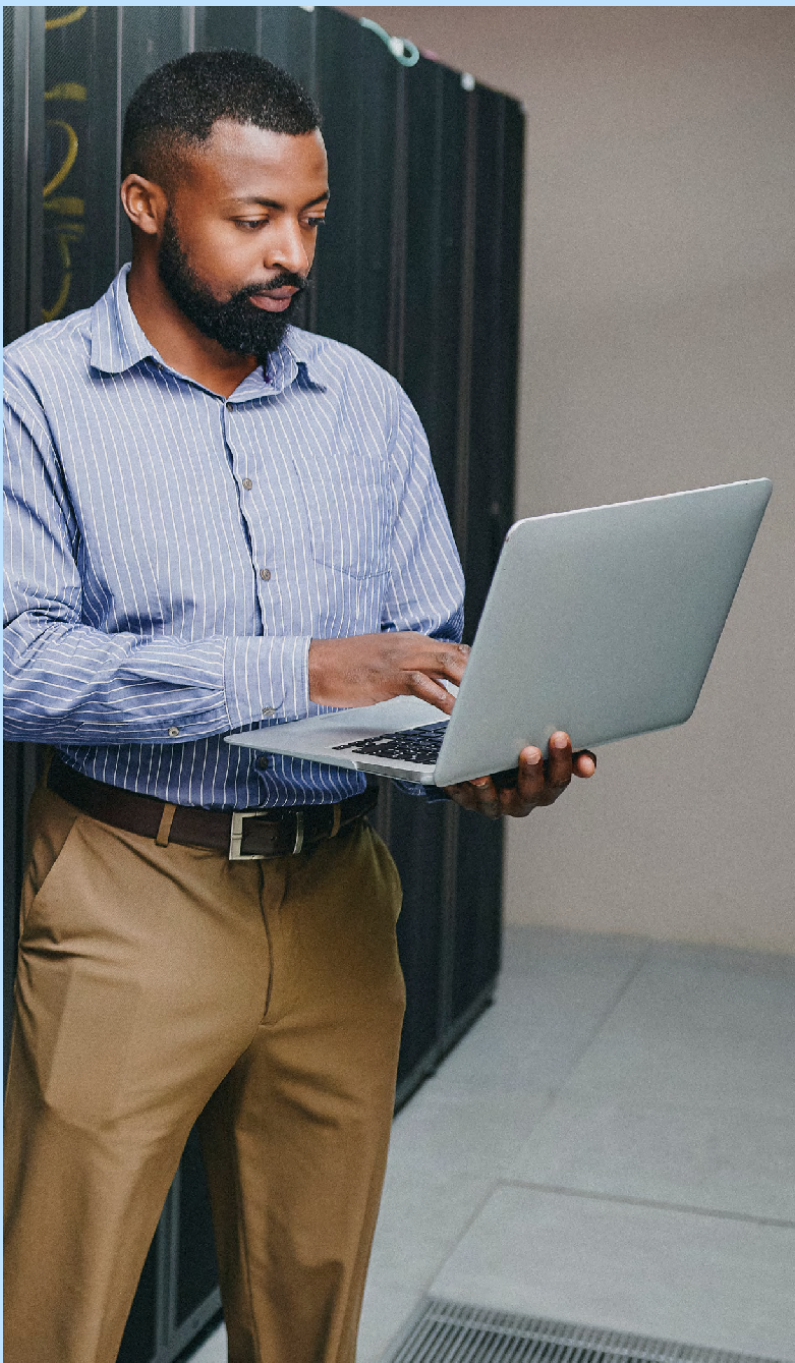
Apply type 3 operation

(i.e. Decrease both of `Y` and `Z` by `1`, then add `A[3]*X*Y*Z` to sum)

`X = 2, Y = 1, Z = 2`

`sum = sum + 3*2*1*2 = 35`

Hence, answer is the final value of `sum` i.e. `sum = 35.`



External Document © 2024 Infosys Limited

Sample 3 : Hard

You are given a tree with **n** nodes rooted at **node 1**. You are also given an array **color** representing the colour of each node in the tree.

A set of nodes is **beautiful** if it satisfies the following conditions:

- All nodes in the set have different colors.
- For any pair of nodes **(u, v)**, either u is the ancestor of v or v is the ancestor of u within the tree.

You're given **q** queries where each query provides an integer **s** representing a node in the tree.

The **answer** to each query is the **maximum size of a beautiful set** that can be formed by selecting nodes from the subtree rooted at **node s**.

Find the **sum of answers to all queries**. Since answer can be large, return it **modulo 10^9+7**.

Notes:

- The parent of **node 1** is **0**.

Input Format

- The first line contains an integer, N, denoting the number of operations.
- The next line contains an integer, X.
- The next line contains an integer,Y.
- The next line contains an integer, Z.
- Each line i of the N subsequent lines (where 1 ≤ i ≤ N) contains an integer describing A[i].
- Each line i of the N subsequent lines (where 1 ≤ i ≤ N) contains an integer describing B[i].

Constraints

- $1 <= N <= 10^3$
- $1 <= X <= 10^3$
- $1 <= Y <= 10^3$
- $1 <= Z <= 10^3$
- $1 <= A[i] <= 10^6$
- $1 <= B[i] <= 10^9$

Sample Input-1:

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

Sample output-1:

```
5
```

Sample Explanation - 1:

Here, `n = 5`

`p = [0, 1, 2, 1, 3]`

`color = [4, 3, 4, 3, 5]`

`q = 3`

`queries = [4, 3, 3]`

for query 1:

`s = 4`, means we need to select beautiful set of maximum size in the subtree of node `4`.

we can select nodes `{4}` to form beautiful set of maximum size.

so, answer for this query is `1`.

for query 2:

`s = 3`, means we need to select beautiful set of maximum size in the subtree of `3`.

we can select nodes `{3, 5}` to form beautiful set of maximum size.

so, answer for this query is `2`.

for query 3:

`s = 3`, means we need to select beautiful set of maximum size in the subtree of 3.

we can select nodes `{3, 5}` to form beautiful set of maximum size.

so, answer for this query is `2`.

Hence, answer is `1 + 2 + 2 = 5`.

Sample input-2:

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

Sample output-2:

```
3
```

Sample Explanation - 2:

Here, `n = 5`

`p = [0, 1, 1, 2, 2]`

`color = [1, 5, 4, 5, 2]`

`q = 3`

`queries = [5, 4, 3]`

for query 1:

`s = 5`, means we need to select beautiful set of maximum size in the subtree of node 5.

we can select nodes `{5}` to form beautiful set of maximum size.

so, answer for this query is 1.

for query 2:

`s = 4`, means we need to select beautiful set of maximum size in the subtree of node 4.

we can select nodes `{4}` to form beautiful set of maximum size.

so, answer for this query is `1`.

for query 3:

`s = 3`, means we need to select beautiful set of maximum size in the subtree of node 3.

we can select nodes {3} to form beautiful set of maximum size.

so, answer for this query is 1.

Hence, answer is `1 + 1 + 1 = 3`.

Sample Input-3:

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

Sample output-3:

```
5
```

Sample Explanation - 3:

Here, `n = 5`

`p = [0, 1, 1, 1, 3]`

`color = [5, 5, 5, 1, 5]`

`q = 4`

`queries = [2, 4, 5, 1]`

for query 1:

`s = 2`, means we need to select beautiful set of maximum size in the subtree of node 2.

we can select nodes `{2}` to form beautiful set of maximum size.

so, answer for this query is `1`.

for query 2:

`s = 4`, means we need to select beautiful set of maximum size in the subtree of 4.

we can select nodes `{4}` to form beautiful set of maximum size.

so, answer for this query is 1.

for query 3:

`s = 5`, means we need to select beautiful set of maximum size in the subtree of 5.

we can select nodes {5} to form beautiful set of maximum size.

so, answer for this query is 1.

for query 4:

`s = 1`, means we need to select beautiful set of maximum size in the subtree of 1.

we can select nodes {1, 4} to form beautiful set of maximum size.

so, answer for this query is `2`.

Hence, answer is `1 + 1 + 1 + 2 = 5`.

© 2024 Infosys Limited, Bengaluru, India. All Rights Reserved. Infosys believes the information in this document is accurate as of its publication date; such information is subject to change without notice. Infosys acknowledges the proprietary rights of other companies to the trademarks, product names and such other intellectual property rights mentioned in this document. Except as expressly permitted, neither this documentation nor any part of it may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, printing, photocopying, recording or otherwise, without the prior permission of Infosys Limited and/ or any named intellectual property rights holders under this document.