

LIBRARY CHECKER

Tác giả: Phan Thành Hưng

January 27, 2026

Tài liệu gồm 5 bài, có 6 trang.

1 SEGTREE - Segment Tree (Basic)

Cho dãy số nguyên a_1, a_2, \dots, a_n . Xử lý q truy vấn thuộc một trong các dạng sau:

- **[1 k x]**: Gán phần tử thứ k thành x .
- **[2 l r]**: Tính tổng các phần tử trong đoạn $[l, r]$.
- **[3 l r]**: Tìm phần tử có giá trị nhỏ nhất trong đoạn $[l, r]$.
- **[4 l r]**: Tìm phần tử có giá trị lớn nhất trong đoạn $[l, r]$.

Dữ liệu vào.

- Dòng đầu tiên gồm hai số nguyên dương n, q ($1 \leq n, q \leq 10^5$).
- Dòng thứ hai gồm n số nguyên a_1, a_2, \dots, a_n ($|a_i| \leq 10^9$).
- q dòng tiếp theo, mỗi dòng chứa một truy vấn.

Dữ liệu ra.

- Với mỗi truy vấn loại 2, 3 hoặc 4, in ra một dòng chứa kết quả tương ứng.

Ví dụ 1

Input

```
8 10
5 -1 3 7 -4 2 6 -8
2 1 8
3 2 5
4 4 7
1 5 10
2 3 6
1 8 4
2 6 8
3 1 4
1 2 -9
4 1 3
```

Output

```
10
-4
7
22
12
-1
5
```

2 SEGLAZY - Segment Tree (Lazy propagation)

Cho dãy số nguyên a_1, a_2, \dots, a_n . Xử lý q truy vấn thuộc một trong các dạng sau:

- **[1 l r k]**: Tăng mỗi phần tử trong đoạn $[l, r]$ lên k đơn vị.
- **[2 l r]**: Tính tổng các phần tử trong đoạn $[l, r]$.
- **[3 l r]**: Tìm phần tử có giá trị nhỏ nhất trong đoạn $[l, r]$.
- **[4 l r]**: Tìm phần tử có giá trị lớn nhất trong đoạn $[l, r]$.

Dữ liệu vào.

- Dòng đầu tiên gồm hai số nguyên dương n, q ($1 \leq n, q \leq 10^5$).
- Dòng thứ hai gồm n số nguyên a_1, a_2, \dots, a_n ($|a_i| \leq 10^9$).
- q dòng tiếp theo, mỗi dòng chứa một truy vấn.

Dữ liệu ra.

- Với mỗi truy vấn loại 2, 3 hoặc 4, in ra một dòng chứa kết quả tương ứng.

Ví dụ 1

Input

```
5 9
-3 5 -2 4 -1
2 1 5
3 1 5
4 1 5
1 2 4 3
2 1 5
3 2 4
4 2 4
1 1 5 -2
2 1 3
```

Output

```
3
-3
5
12
1
7
-1
```

3 SEGPERs - Persistent Segment Tree

Cho dãy số nguyên a_1, a_2, \dots, a_n . Ban đầu, tồn tại phiên bản 1 của dãy, chính là dãy ban đầu. Mỗi truy vấn cập nhật sẽ tạo ra một phiên bản mới của dãy.

Xử lý q truy vấn thuộc một trong các dạng sau:

- **[1 v i k]**: Từ phiên bản v , tạo ra một phiên bản mới bằng cách tăng phần tử thứ i lên k đơn vị ($|k| \leq 10^9$).
- **[2 v l r]**: Tính tổng các phần tử trong đoạn $[l, r]$ của phiên bản v .
- **[3 v l r]**: Tìm giá trị nhỏ nhất trong đoạn $[l, r]$ của phiên bản v .
- **[4 v l r]**: Tìm giá trị lớn nhất trong đoạn $[l, r]$ của phiên bản v .

Các phiên bản được đánh số liên tiếp từ 1 theo thứ tự xuất hiện (các truy vấn loại 1).

Dữ liệu vào.

- Dòng đầu tiên gồm hai số nguyên dương n, q ($1 \leq n, q \leq 10^5$).
- Dòng thứ hai gồm n số nguyên a_1, a_2, \dots, a_n ($|a_i| \leq 10^9$).
- q dòng tiếp theo, mỗi dòng chứa một truy vấn như mô tả ở trên.

Dữ liệu ra.

- Với mỗi truy vấn loại 2, 3 hoặc 4, in ra một dòng chứa kết quả tương ứng.

Ví dụ 2

Input

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

Output

```
6
9
-3
4
3
-5
7
```

4 SEGPERSLAZY - Lazy Persistent Segment Tree

Cho dãy số nguyên a_1, a_2, \dots, a_n . Ban đầu, tồn tại phiên bản 1 của dãy, chính là dãy ban đầu. Mỗi truy vấn cập nhật sẽ tạo ra một phiên bản mới của dãy.

Xử lý q truy vấn thuộc một trong các dạng sau:

- **[1 v l r k]**: Từ phiên bản v , tạo ra một phiên bản mới bằng cách tăng mỗi phần tử trong đoạn $[l, r]$ lên k đơn vị.
- **[2 v l r]**: Tính tổng các phần tử trong đoạn $[l, r]$ của phiên bản v .
- **[3 v l r]**: Tìm giá trị nhỏ nhất trong đoạn $[l, r]$ của phiên bản v .
- **[4 v l r]**: Tìm giá trị lớn nhất trong đoạn $[l, r]$ của phiên bản v .

Các phiên bản được đánh số liên tiếp từ 1 theo thứ tự xuất hiện (các truy vấn loại 1).

Dữ liệu vào.

- Dòng đầu tiên gồm hai số nguyên dương n, q ($1 \leq n, q \leq 10^5$).
- Dòng thứ hai gồm n số nguyên a_1, a_2, \dots, a_n ($|a_i| \leq 10^9$).
- q dòng tiếp theo, mỗi dòng chứa một truy vấn như mô tả ở trên.

Dữ liệu ra.

- Với mỗi truy vấn loại 2, 3 hoặc 4, in ra một dòng chứa kết quả tương ứng.

Ví dụ 2

Input

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

Output

```
6
18
-3
7
-3
-5
7
```

5 WAVELET - Wavelet Tree

Cho dãy số nguyên a_1, a_2, \dots, a_n . Hãy xử lý q truy vấn thuộc một trong các dạng sau:

- **[1 l r k]**: Phần tử lớn thứ k trong đoạn $[l, r]$. Nếu không tồn tại, in ra NO.
- **[2 l r x]**: Số phần tử $\leq x$ trong đoạn $[l, r]$.
- **[3 l r x]**: Tổng các phần tử $\leq x$ trong đoạn $[l, r]$.
- **[4 l r x]**: Số phần tử $= x$ trong đoạn $[l, r]$.

Dữ liệu vào.

- Dòng đầu tiên gồm hai số nguyên dương n, q ($1 \leq n, q \leq 10^5$).
- Dòng thứ hai gồm n số nguyên a_1, a_2, \dots, a_n ($|a_i| \leq 10^6$).
- q dòng tiếp theo, mỗi dòng chứa một truy vấn.

Dữ liệu ra.

- Gồm q dòng, mỗi dòng trả lời kết quả của truy vấn tương ứng.

Ví dụ 1

Input

```
6 8
-5 3 -2 7 -2 4
1 1 6 3
2 1 6 -2
3 1 6 0
4 1 6 -2
1 2 5 5
2 2 5 10
3 2 5 3
4 3 6 7
```

Output

```
3
3
-9
2
NO
4
-1
1
```