

Task 2. Lider

Lider multizbioru (zbiór, w którym mogą powtarzać się elementy) to taki element, który występuje częściej niż pozostałe elementy razem wzięte. Innymi słowy, lider występuje więcej niż połowa razy. Zauważ, że niektóre multizbiory nie mają wogóle lidera.

Dana jest tablica n dodatnich liczb całkowitych $a[1], a[2], \dots, a[n]$. Podtablicą tablicy będziemy nazywali jej spójny podciąg: $a[l], a[l+1] \dots a[r]$, gdzie $1 \leq l \leq r \leq n$.

Rozważamy m zapytań dwóch typów:

- 1) Aktualizuj („Update”) p q . W tym zapytaniu $a[p]$ przypisujemy wartość q .
- 2) Zapytanie („Query”) p q , gdzie $p \leq q$. Niech $b[1]=a[p], b[2]=a[p+1] \dots b[q-p+1]=a[q]$. Dla tego zapytania, powinnaś (pozdrowanie dla Zuzi), a Wy (pozdrowanie dla chłopaków) powinniście policzyć sumę, złożoną z sum postaci: $i \cdot$ (liczba podtablic b , które mają lidera i) dla wszystkich takich i , że i jest liderem przynajmniej jednej podtablicy b . Wynik podaj modulo 998244353.

Wejście:

Pierwsza linia standardowego wejścia zawiera jedną liczbę całkowitą n . Druga linia standardowego wejścia zawiera n liczb całkowitych – wartości kolejnych elementów tablicy a . Trzecia linia standardowego wejścia zawiera liczbę m . W kolejnych m liniach opisano kolejne zapytania. Każde zapytanie podane jest w zaszyfrowanej formie i jest opisane za pomocą trzech liczb: l, r i t .

Odszyfrowanie: niech `last_output` będzie ostatnią liczbą wypisaną na wyjście przez Twój program (lub 0, jeśli jeszcze nic nie wypisał).

Obliczenie typu zapytania jest następujące: $\text{type} = ((t + \text{last_output}) \bmod 2) + 1$.

Jeśli $\text{type} = 1$, wtedy rozważamy zapytanie "Update" z $p = ((l + \text{last_output}) \bmod n) + 1$ i $q = ((r + \text{last_output}) \bmod 100\,000\,000) + 1$.

Jeśli $\text{type} = 2$, wtedy rozważamy zapytanie "Query" z $p = ((l + \text{last_output}) \bmod n) + 1$ i $q = ((r + \text{last_output}) \bmod n) + 1$.

Wyjście

Dla każdego zapytania typu 2., wypisz w oddzielnej linii odpowiedź na zapytanie.

Ograniczenia

$$1 \leq n \leq 200\,000$$

Dla każdej liczby x w tablicy a zachodzi, że $1 \leq x \leq 100\,000\,000$.

$$1 \leq m \leq 100$$

$$1 \leq l, r, t \leq 100\,000\,000$$

Podzadania

Podzadanie	Punkty	Dodatkowe ograniczenia
1	10	$n \leq 100, m \leq 50$
2	15	$n \leq 1\ 000, m \leq 50$
3	10	$n \leq 10\ 000, m \leq 50$
4	10	$n \leq 50\ 000, m \leq 5$
5	20	$n \leq 65\ 000, m \leq 50$
6	35	brak

Punkty za podzadanie zostaną przyznane wtedy i tylko wtedy, jeśli program poprawnie rozwiąże wszystkie testy dla tego podzadania.

Przykładowe wejście

```
4
1 2 2 1
3
4 3 1
2 99999990 2
4 2 1
```

Przykładowe wyjście

```
12
6
```

Wyjaśnienie

Po odszyfrowaniu, pierwsze zapytanie jest postaci „query, p=1, q=4”. Wtedy w a[1], a[2], a[3], a[4] mamy dwie podtablice z liderem 1 oraz 5 podtablic z liderem 2, zatem odpowiedzią jest $2*1+5*2=12$.

Drugie zapytanie jest postaci: „update, p=3, q=3”. Po tym zapytaniu mamy tablicę: 1, 2, 3, 1.

Trzecie zapytanie jest postaci: „query, p=1, q=3”. Wtedy w a[1], a[2], a[3] mamy jedną podtablicę z liderem 1, jedną podtablicę z liderem 2 i jedną podtablicę z liderem 3.