

Dostępna pamięć: 256MB

Podział zbioru II

Zadanie polega na opracowaniu i zaimplementowaniu struktury danych umożliwiającej wykonywanie następujących operacji na podziale zbioru $\{1, 2, \dots, n\}$:

- `init()`: inicjuje podział singletonowy:

$$\{\{1\}, \{2\}, \dots, \{n\}\};$$

- `find(x)`: znajduje sumę, element najmniejszy, element największy oraz medianę zbioru podziału zawierającego element `x`;
- `union(x, y)`: łączy zbiory zawierające element `x` oraz `y`, jeśli są to różne zbiory. Jeśli `x` i `y` należą do tego samego zbioru, operacja nic nie robi.

Wejście

Na wejściu znajdują się dwie liczby całkowite n i m ($1 \leq n \leq 200\,000$, $2 \leq m \leq 300\,000$) oznaczające liczbę elementów zbioru oraz liczbę operacji. Pierwszą wykonywaną operacją jest operacja `init()`. W $m - 1$ kolejnych wierszach znajdują się opisy operacji:

- `find x` oznacza operację `find(x)` ($1 \leq x \leq n$),
- `union x y` oznacza operację `union(x, y)` ($1 \leq x, y \leq n$).

Wyjście

Dla każdej operacji `find(x)` proszę wypisać trzy liczby całkowite – sumę elementów, element najmniejszy, element największy oraz medianę zbioru podziału zawierającego element `x`. Jeśli liczba elementów zbioru jest parzysta, proszę podać nie mniejszy z dwóch środkowych elementów.

Przykład

| Wejście | Wyjście |
|-----------|----------|
| 5 8 | 4 4 4 4 |
| find 4 | 7 2 5 5 |
| union 2 5 | 4 4 4 4 |
| find 5 | 11 1 5 3 |
| union 1 3 | |
| union 2 3 | |
| find 4 | |
| find 2 | |