

Opis

Dane: $n > 1$ oraz tablica liczb całkowitych: $a[0], \dots, a[n-1]$. Inwersją w tablicy a nazywamy każdą parę liczb (i, j) : $i < j$ dla których $a[i] > a[j]$.

Korzystając z metody „dziel i zwyciężaj” napisz program w Javie zawierający rekurencyjną funkcję `long numInversion(long a[], int L, int R)`, działającą w czasie $O(n \log n)$, która wyznacza liczbę inwersji w tablicy $a[L, R]$.

Wejście

Dane do programu wczytywane są ze standardowego wejścia (klawiatury) zgodnie z poniższą specyfikacją.

- ❑ Pierwszą podawaną wartością jest dodatnia liczba całkowita oznaczająca ilość zestawów danych, po której na wejściu pojawią się zestawy danych w ilości równej wczytanej liczbie.
- ❑ Każdy zestaw danych zawiera:
 - Dodatnią liczbę całkowitą z zakresu od 1 do 2^{15} oznaczającą ilość danych wczytywanego zestawu.
 - Zasadnicze dane zestawu w ilości równej poprzednio wczytanej wartości, będące liczbami całkowitymi z zakresu od -2^{48} do $+2^{48}$.

Wyjście

Dla każdego zestawu danych wypisz ilość inwersji w zadanej tablicy.

Wymagania implementacyjne

Jak w Programie 1.

Przykład danych

wejście:	wyjście:
4	0
10	9
1 2 3 4 5 6 7 8 9 10	45
10	0
10 1 2 3 4 5 6 7 8 9	
10	
10 9 8 7 6 5 4 3 2 1	
10	
0 0 0 0 0 0 0 0 0 0	