


| | | |
|-----------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------|-----------------------------------------|
|  | <p style="text-align: center;">Metody programowania 2021/2022 Selekcja piosenek</p> | <p style="text-align: center;">P_08</p> |
|-----------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------|-----------------------------------------|

Opis

Pan Michał Meloman ma dla Ciebie kolejne zadanie. Dysponuje on tablicą, w której pod indeksem oznaczającym numer piosenki na płycie przechowuje informację o liczbie głosów oddanych na tę piosenkę w głosowaniu Wieczornej Listy Przebojów. Pan Meloman chciałby wiedzieć, ile głosów zdobyła piosenka zajmująca określoną przez niego pozycję w notowaniu licząc od końca. Dodatkowo może zdarzyć się tak, że kilka piosenek zdobyło tyle samo głosów. Twoim zadaniem będzie więc napisanie rekurencyjnej funkcji znajdującej k -ty co do wielkości element tablicy n -elementowej $a[0], \dots, a[n-1]$ w porządku niemalejącym, licząc od najmniejszego. Wartość $k = 1$ odpowiada najmniejszemu elementowi tablicy, czyli piosence która zdobyła najmniej głosów.

Wejście

Pierwszą podawaną wartością będzie dodatnia liczba całkowita z ($1 \leq z \leq 100$) oznaczająca liczbę zestawów danych. Każdy zestaw danych zawiera kolejno:

1. liczbę całkowitą n ($1 \leq n \leq 10^6$), oznaczającą długość tablicy,
2. n liczb całkowitych a_i ($0 \leq a_i \leq 10^6$), będących elementami tablicy,
3. liczbę całkowitą m ($1 \leq m \leq 100$), oznaczającą liczbę zapytań o określony element,
4. m liczb całkowitych k_j ($1 \leq k_j \leq 10^6$), z których każda oznacza k_j - ty element tablicy, który należy znaleźć.

Wyjście

Dla każdego zestawu danych program wypisuje m kolejnych linii, z których każda zawiera jedną z dwóch opcji:

- liczbę k_i oraz wartość k_i - tego co do wielkości elementu tablicy w porządku niemalejącym, oddzielone spacją - jeśli taki element istnieje,
- lub
- liczbę k_i i słowo „brak” oddzielone spacją - w przeciwnym przypadku.

Wymagania implementacyjne

1. Główna funkcja programu na której opiera się cały algorytm ma być funkcją rekurencyjną i działać w oparciu o algorytm magicznych piątek przedstawiony na wykładzie.
2. W przypadku pesymistycznym program ma mieć liniową złożoność czasową w zależności od liczby danych.
3. Program ma mieć stałą złożoność pamięciową bez wliczania stosu wywołań funkcji (nie może rezerwować dodatkowych tablic ani list, ma działać „w miejsku”).



Inne uwagi

1. W pierwszej linii program powinien zawierać komentarz:
// Nazwisko i imię – nr grupy
2. Jedyną biblioteką którą można zaimportować jest ta obsługująca wczytywanie wejścia:
import java.util.Scanner;
a wczytywanie ma się odbywać przez pojedynczą zmienną zainicjalizowaną jako:
new Scanner(System.in);
3. Program powinien zawierać komentarze w formie podanej w punkcie 3 i 4 aktualnego Regulaminu zaliczania programów na BaCy oraz ocenę złożoności rozwiązania wraz z uzasadnieniem.

Przykład danych

| Wejście: | Wyjście: |
|---------------------|----------|
| 3 | 1 1 |
| 5 | 2 2 |
| 1 2 3 4 5 | 3 3 |
| 3 | 2 3 |
| 1 2 3 | 5 5 |
| 5 | 1 3 |
| 5 3 4 4 3 | 3 4 |
| 5 | 4 4 |
| 2 5 1 3 4 | 1 1 |
| 10 | 10 1 |
| 1 1 1 1 1 1 1 1 1 1 | 0 brak |
| 5 | 20 brak |
| 1 10 0 20 11 | 11 brak |