	<p>Metody programowania 2016/2017</p> <p>Porządek w tablicy</p>	<p>P02</p>
---	---	------------

Opis

Dana jest uporządkowana niemalejąco tablica liczb całkowitych $a[n]$, przy czym $2 < n < 2^{15}$.

Napisz w Javie efektywny pamięciowo i obliczeniowo program działający na tablicy realizujący następujące operacje:

1. Obliczenia liczby elementów w tablicy równych zadanej wartości o złożoności pesymistycznej $O(\log n)$.
2. Obliczenia indeksu elementu tablicy równego zadanej wartości metodą wyszukiwania interpolacyjnego.
3. Usuwania duplikatów o złożoności $O(n)$.

Wejście


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

- ❑ Pierwszą podawaną wartością jest dodatnia liczba całkowita z zakresu od 1 do 2^{15} , 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 $+2^{48}$, podanymi z zachowaniem *niemalejącego uporządkowania*.
 - Dodatnią liczbę całkowitą z zakresu od 1 do 2^{15} oznaczającą ilość zapytań o szukaną wartość
 - Liczby całkowite z zakresu od -2^{48} do $+2^{48}$ określające szukane wartości stanowiące przedmiot zapytań w ilości równej poprzednio wczytanej liczbie.

Wyjście

Dla każdej zestawu danych, dla kolejnych wartości będących przedmiotem zapytania wypisz linie zawierające:

- Pierwsza: pary liczb w postaci: (i j) oddzielone spacją przy czym:
 - i – jest liczbą elementów tablicy o zadanej wartości,
 - j – jest równy -1 jeśli tablica nie zawiera danej wartości, w przeciwnym przypadku jest indeksem elementu tablicy, który jest równy zadanej wartości, wyszukanej metodą interpolacyjną. Przy czym jeśli wszystkie elementy tablicy są równe zadanej wartości - j jest równe 0.

	<p>Metody programowania 2016/2017</p> <p>Porządek w tablicy</p>	<p>P02</p>
---	---	------------

- W kolejnych – wypisz co najwyżej 200 początkowych elementów tablicy po usunięciu duplikatów, oddzielone spacją, po 50 elementów w jednym wierszu.
- Przy czym dla każdego zestawu wypisywanie wyników rozpoczyna się w nowej linii.

Wymagania implementacyjne

Jak w Programie 1.

Przykład danych

<p><u>test1.in:</u></p> <pre>1 12 -1 1 2 2 2 3 5 5 7 7 9 9 10 1 2 3 -1 4 9 5 6 7 8</pre> <p><u>test1.out:</u></p> <pre>(1 1) (3 3) (1 5) (1 0) (0 -1) (2 10) (2 6) (0 -1) (2 8) (0 -1) -1 1 2 3 5 7 9</pre>
<p><u>test2.in:</u></p> <pre>1 10 1 2 2 2 2 2 3 3 3 3 5 1 2 3 4 0</pre> <p><u>test2.out:</u></p> <pre>(1 0) (5 4) (4 9) (0 -1) (0 -1) 1 2 3</pre>
<p><u>test3.in:</u></p> <pre>1 10 0 0 0 0 0 0 0 0 0 1 4 0 -1 1 2</pre> <p><u>test3.out:</u></p> <pre>(9 0) (0 -1) (1 9) (0 -1) 0 1</pre>
<p><u>test4.in:</u></p> <pre>1 10 1 1 1 1 1 1 1 1 1 1 3 1 0 2</pre> <p><u>test4.out:</u></p> <pre>(10 0) (0 -1) (0 -1) 1</pre>