



## B. Array

Link submit: <http://codeforces.com/problemset/problem/224/B>

Solution:

C++: <https://ideone.com/DslkHp>

Java: <https://ideone.com/E07JWy>

Python: <https://ideone.com/23qcAu>

**Tóm tắt đề:** Cho bạn mảng  $a$  gồm  $n$  phần tử. Các vị trí được đánh dấu từ 1 đến  $n$ . Cho bạn một số  $k$ .

Hãy tìm một đoạn  $[l, r]$  ( $1 \leq l \leq r \leq n$ ) sao cho trong đoạn  $[l, r]$  này chứa  $k$  phần tử phân biệt. Khi tìm đoạn  $[l, r]$  này thì chiều dài của đoạn này không nhất thiết phải là nhỏ nhất nhưng không được phép có đoạn nào khác tồn tại bên trong nó mà vẫn thỏa điều kiện yêu cầu đề bài. Nghĩa là không có đoạn  $[x, y]$  nào  $1 \leq x \leq y \leq r \leq n$  thỏa mãn tính chất trên.

### Input

Dòng đầu tiên chứa 2 số  $n, k$  ( $1 \leq n, k \leq 10^5$ ).

Dòng tiếp theo chứa  $n$  số nguyên  $a_1, a_2, \dots, a_n$  là các phần tử trong mảng  $a$  ( $1 \leq a_i \leq 10^5$ ).

### Output

In ra kết quả 2 số  $l, r$  là đoạn cần tìm, nếu không tìm được đoạn nào thỏa mãn thì in ra "-1 -1".

8 3 1 1 2 2 3 3 4 5	2 5
------------------------	-----

**Giải thích:** Vị trí số đầu tiên bắt đầu là vị trí 1 (không phải 0 giống trong mảng).

Ở ví dụ này ta thấy đoạn 2 5 tuy không phải là đoạn có độ dài nhỏ nhất nhưng là đoạn có tối thiểu 3 giá trị phân biệt. Và trong đoạn 2 5 này không tồn tại một đoạn nào nhỏ hơn thỏa mãn điều kiện trên nên đoạn 2 5 là đoạn cần tìm.

### Hướng dẫn giải:

Đầu tiên là chạy tìm đoạn dài nhất có thể chứa các số phân biệt bằng  $k$ . Sau đó chạy từ dưới lên để rút ngắn đoạn đó lại để không có còn nào nhỏ hơn đoạn đó thỏa mãn đề bài. Như vậy là ta tìm được kết quả bài toán.

Do đề bài không yêu cầu tìm đoạn ngắn nhất chứa k số phân biệt mà chỉ cần tìm đoạn nào đó chứa đủ k số và không có đoạn con nào trong nó thỏa mãn đề bài là được.

**Độ phức tạp:**  $O(n)$  với n số lượng phần tử trong dãy.

Big-O Coding