**F - Pearls in a Row**

**Time Limit:**2000MS     **Memory Limit:**262144KB     **64bit IO Format:**%I64d & %I64u

[Submit](http://vjudge.net/contest/142389" \l "problem/javascript:void(0)) [Status](http://vjudge.net/contest/142389" \l "status//F/0)

**Description**

There are *n* pearls in a row. Let's enumerate them with integers from 1 to *n* from the left to the right. The pearl number *i* has the type *ai*.

Let's call a sequence of consecutive pearls a segment. Let's call a segment good if it contains two pearls of the same type.

Split the row of the pearls to the maximal number of good segments. Note that each pearl should appear in exactly one segment of the partition.

As input/output can reach huge size it is recommended to use fast input/output methods: for example, prefer to use scanf/printf instead ofcin/cout in C++, prefer to use BufferedReader/PrintWriter instead of Scanner/System.out in Java.

**Input**

The first line contains integer *n* (1 ≤ *n* ≤ 3·105) — the number of pearls in a row.

The second line contains *n* integers *ai* (1 ≤ *ai* ≤ 109) – the type of the *i*-th pearl.

**Output**

On the first line print integer *k* — the maximal number of segments in a partition of the row.

Each of the next *k* lines should contain two integers *lj*, *rj* (1 ≤ *lj* ≤ *rj* ≤ *n*) — the number of the leftmost and the rightmost pearls in the*j*-th segment.

Note you should print the correct partition of the row of the pearls, so each pearl should be in exactly one segment and all segments should contain two pearls of the same type.

If there are several optimal solutions print any of them. You can print the segments in any order.

If there are no correct partitions of the row print the number "-1".

**Sample Input**

**Input**

5  
1 2 3 4 1

**Output**

1  
1 5

**Input**

5  
1 2 3 4 5

**Output**

-1

**Input**

7  
1 2 1 3 1 2 1

**Output**

2  
1 3  
4 7

一行一列的珍珠

时间限制：2000ms内存限制：262144kb 64bit IO格式：%I64d & % i64u

提交状态

描述；形容；种类；类型；

一行有n个珍珠。让我们列举他们的整数从1到n从左到右。珍珠号我有型人工智能。

让我们把一个连续的珍珠序列。如果它包含两个相同类型的珍珠，让我们打电话给一段不错的。

把珍珠的行分割为好的部分的最大数。请注意，每一个珍珠应该出现在一个分区的分区。

作为输入/输出可以达到巨大的尺寸建议使用快速输入/输出的方法：例如，喜欢使用scanf / printf代替宫颈/ cout在C++，喜欢使用BufferedReader / PrintWriter所EAD扫描仪/ System.out java。

输入，投入；输入电路；输入端；输入的数据；把…输入电脑；[自][电子]输入；输入，给料；

第一行包含整数N（1≤N≤3·105）-在一排珍珠的数量。

第二行包含N个整数ai（1≤AI≤109）–第i珍珠的类型。

产量；输出；作品；[计]输出信号；输出；

在第一行打印整数k -在一个分区的分区的最大数量。

接下来的每一个K线应包含两个整数LJ，RJ（1≤LJ≤RJ≤N）-最左边的数和段最右边的珍珠效果。

注意，你应该打印的珍珠行正确的分区，所以每一个珍珠应该在一个段，所有部分应包含两个相同类型的珍珠。

如果有几个最佳的解决方案打印他们的任何。您可以以任何顺序打印这些段。

如果没有正确的分区的行打印的号码“- 1”。

样本输入，样本值输入，抽样输入；

输入，投入；输入电路；输入端；输入的数据；把…输入电脑；[自][电子]输入；输入，给料；

543211

产量；输出；作品；[计]输出信号；输出；

1 1 5

输入，投入；输入电路；输入端；输入的数据；把…输入电脑；[自][电子]输入；输入，给料；

1 2 3 4 5

产量；输出；作品；[计]输出信号；输出；

- 1

输入，投入；输入电路；输入端；输入的数据；把…输入电脑；[自][电子]输入；输入，给料；

11712321

产量；输出；作品；[计]输出信号；输出；

1 2 4 7 3