Square-Ten Tree

H by vasily_makarov

| Problem | Submissions | Leaderboard | Discussions |

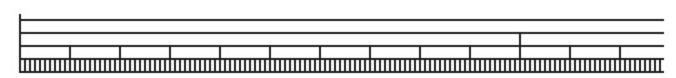The square-ten tree decomposition of an array is defined as follows:

- The lowest (0th) level of the square-ten tree consists of single array elements in their natural order.

- The kth level (starting from 1) of the square-ten tree consists of subsequent array subsegments of length $10^{2^{k-1}}$ in their natural order. Thus, the 1st level contains subsegments of length $10^{2^{1-1}} = 10$, the 2nd level contains subsegments of length $10^{2^{2-1}} = 100$, the 3rd level contains subsegments of length $10^{2^{3-1}} = 10000$, etc.

In other words, every k^{th} level (for every $k \geq 1$) of square-ten tree consists of array subsegments indexed as:

$$\left[1,\ 10^{2^{k-1}}\right], \left[10^{2^{k-1}}+1,\ 2 \cdot 10^{2^{k-1}}\right], \dots, \left[i \cdot 10^{2^{k-1}}+1,\ (i+1) \cdot 10^{2^{k-1}}\right], \dots$$

Level 0 consists of array subsegments indexed as $[1,\ 1], [2,\ 2], \dots, [i,\ i], \dots$.

The image below depicts the bottom-left corner (i.e., the first 128 array elements) of the table representing a square-ten tree. The levels are numbered from bottom to top:

Task

Given the borders of array subsegment $[L, R]$, find its decomposition into a minimal number of nodes of a square-ten tree. In other words, you must find a subsegment sequence $[l_1, r_1], [l_2, r_2], \dots, [l_m, r_m]$ such as $l_{i+1} = r_i + 1$ for every $1 \leq i < m$, $l_1 = L$, $r_m = R$, where every $[l_i, r_i]$ belongs to any of the square-ten tree levels and m is minimal amongst all such variants.

Input Format

The first line contains a single integer denoting L.
The second line contains a single integer denoting R.

Constraints

- $1 \leq L \leq R \leq 10^{10^6}$

- The numbers in input do not contain leading zeroes.

Output Format

As soon as array indices are too large, you should find a sequence of m square-ten tree level numbers, $s_1, s_2, \dots, s_m$, meaning that subsegment $[l_i, r_i]$ belongs to the s_i^{th} level of the square-ten tree.

Print this sequence in the following compressed format:

- On the first line, print the value of n (i.e., the compressed sequence block count).

- For each of the n subsequent lines, print 2 space-separated integers, t_i and c_i ($t_i \geq 0, c_i \geq 1$), meaning that the number t_i appears consequently c_i times in sequence s. Blocks should be listed in the order they appear in the sequence. In other words, $s_1, s_2, \dots, s_{c_1}$ should be equal to t_1, $s_{c_1+1}, s_{c_1+2}, \dots, s_{c_1+c_2}$ should be equal to t_2, etc.

Thus $\sum_{i=1}^{n} c_i = m$ must be true and $t_i \neq t_{i+1}$ must be true for every $1 \leq i < n$. All numbers should be printed without leading zeroes.

Sample Input

```
1
10
```

Sample Output

```
1
1 1
```

Explanation

Segment $[1, 10]$ belongs to level 1 of the square-ten tree.

Submissions: 80

Max Score: 60

Difficulty: Moderate

More

Current Buffer (saved locally, editable)   Java 8

```java
1  import java.io.*;
2  import java.util.*;
3
4  public class Solution {
5
6      public static void main(String[] args) {
7          /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class should be named Solution. */
8      }
9  }
```

Line: 1 Col: 1

Upload Code as File   ☐ Test against custom input   Run Code   Submit Code

Join us on IRC at #hackerrank on freenode for hugs or bugs.

Contest Calendar | Blog | Scoring | Environment | FAQ | About Us | Support | Careers | Terms Of Service | Privacy Policy | Request a Feature