

Candidate Report: Anonymous



TASKS DETAILS



Task description

A DNA sequence can be represented as a string consisting of the letters A, C, G and T, which correspond to the types of successive nucleotides in the sequence. Each nucleotide has an impact factor, which is an integer. Nucleotides of types A, C, G and T have impact factors of 1, 2, 3 and 4, respectively. You are going to answer several queries of the form: What is the minimal impact factor of nucleotides contained in a particular state of the form: What is the minimal impact factor of nucleotides contained in a particular part of the given DNA sequence?

The DNA sequence is given as a non-empty string $S = S[0]S[1] \dots S[H-1]$ consisting of N characters. There are M queries, which are given in non-empty arrays P and Q, each consisting of M integers. The K-th query (0 $\le K$ N) requires you to find the minimal impact factor of nucleotides contained in the DNA sequence between positions P[K] and Q[K] (inclusive).

For example, consider string S = CAGCCTA and arrays P, Q such that:

```
P[0] = 2 Q[0] = 4

P[1] = 5 Q[1] = 5

P[2] = 0 Q[2] = 6
```

The answers to these M = 3 queries are as follows:

- The part of the DNA between positions 2 and 4 contains nucleotides G
- The part of the UNA between positions 2 and 4 contains nucleotides 6 and C (twice), whose impact factors are 3 and 2 respectively, so the answer is 2.
 The part between positions 5 and 5 contains a single nucleotide 7, whose impact factor is 4, so the answer is 4.
 The part between positions 0 and 6 (the whole string) contains all nucleotides, in particular nucleotide A whose impact factor is 1, so the answer is 1.

```
class Solution { public int[] solution(String S, int[] P, int[] Q); }
```

that, given a non-empty string S consisting of N characters and two non-empty arrays P and Q consisting of M integers, returns an array consisting of M integers specifying the consecutive answers to all queries.

Result array should be returned as an array of integers.

For example, given the string S = CAGCCTA and arrays P, Q such that:

```
P[0] = 2 Q[0] = 4

P[1] = 5 Q[1] = 5

P[2] = 0 Q[2] = 6
```

the function should return the values [2, 4, 1], as explained above

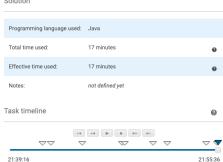
Write an efficient algorithm for the following assumptions:

- N is an integer within the range [1..100,000];

- M is an integer within the range [1..50,000]:
 each element of arrays P, Q is an integer within the range [0..N 1]:
 $P[K] \le Q[K]$, where $Q \le K < M$:
 string S consists only of upper-case English letters A, C, G, π .

Copyright 2009-2019 by Codility Limited. All Rights Reserved. Unauthorized copying, publication or disclosure

Solution



100	de: 21:55:36 UTC, java, final, score: show code in pop-u
1	<pre>import java.util.*;</pre>
2	
3	class Solution {
4	<pre>public int[] solution(String S, int[] P, int[] Q) {</pre>
5	<pre>char[] cA = S.toCharArray();</pre>
6	<pre>int[] pSumA = getPSum('A', cA);</pre>
7	<pre>int[] pSumC = getPSum('C', cA);</pre>
8	<pre>int[] pSumG = getPSum('G', cA);</pre>
9	des escale and
1	int start, end;
12	<pre>int[] result = new int[P.length]; for (int i = 0; i < P.length; i++) {</pre>
3	start = P[i];
4	end = Q[i] + 1;
5	ena - g[1] · 1)
16	<pre>if (pSumA[end] - pSumA[start] > 0) result[i] = 1;</pre>
7	<pre>else if (pSumC[end] - pSumC[start] > 0) result[i] = 2;</pre>
.8	<pre>else if (pSumG[end] - pSumG[start] > 0) result[i] = 3;</pre>
9	else result[i] = 4;
0)
1	,
2	return result;
23)
4	
5	<pre>private int[] getPSum(char c, char[] cA) {</pre>
26	<pre>int[] pSum = new int[cA.length + 1];</pre>
27	
8.	if (cA[0] == c) {
29	pSum[0] = 1;
10) else {
1	pSum[0] = 0;
3	}
4	for (int i = 0, i < a) learth, its (
15	<pre>for (int i = 0; i < cA.length; i++) { if (cA[i] == c) {</pre>
36	pSum[i + 1] = pSum[i] + 1;
37	else {
88	pSum[i + 1] = pSum[i];
19	}
10	, '
11	,
2	return pSum;
3)
4)

Analysis summary

The solution obtained perfect score

Analysis 👩



▶ extreme_large all max ranges