

CSES Problem Set

Bit Substrings

TASK | SUBMIT | RESULTS | STATISTICS

Submission details

Task:	<u>Bit Substrings</u>
Sender:	Rodry
Submission time:	2021-12-23 09:20:11
Language:	C++17
Status:	READY
Result:	TIME LIMIT EXCEEDED

Test results ▲

test	verdict	time	
#1	ACCEPTED	0.01 s	<u>>></u>
#2	ACCEPTED	0.01 s	<u>>></u>
#3	ACCEPTED	0.01 s	<u>>></u>
#4	TIME LIMIT EXCEEDED		<u>>></u>
#5	TIME LIMIT EXCEEDED		<u>>></u>
#6	TIME LIMIT EXCEEDED		<u>>></u>
#7	TIME LIMIT EXCEEDED		<u>>></u>
#8	TIME LIMIT EXCEEDED		<u>>></u>

Code ▲

1	<pre>#include <bits stdc++.h=""></bits></pre>
2	<pre>using namespace std;</pre>
3	
4	<pre>void printArray(int A[],int n){</pre>
5	<pre>for(int i=0;i<n;i++)< pre=""></n;i++)<></pre>
6	cout< <a[i]<<" ";<="" td=""></a[i]<<">

Additional Problems

Grid Puzzle II	_
Empty String	_
Grid Paths	_
Bit Substrings	X
Reversal Sorting	_
Counting Reorders	_
Book Shop II	_
Network Breakdown	_

Your submissions

2021-12-23 09:20:11	X
2021-12-23 09:17:15	X
2021-12-23 09:16:35	X
2021-12-23 09:15:19	X

```
7 | }
 8
 9 int Bit_Substrings(int A[],int n){
10
        int B[n+1];
        B[0]=0;
11
12
13
        for(int i=1;i<n+1;i++){</pre>
14
            B[i]=B[i-1]+A[i-1];
15
16
17
        int ANS[n+1];
18
        for(int i=0;i<n+1;i++) ANS[i]=0;</pre>
19
        for(int size=0;size<n+1;size++){</pre>
20
21
            for(int i=1,j=i+size;j<n+1;i++,j++){</pre>
22
                int unos=0;
23
                if(B[i]!=B[i-1]) unos++;
24
                unos+=B[j]-B[i];
25
                ANS[unos]++;
26
27
28
29
30
        printArray(ANS,n+1);
31
32
        return 0;
33 }
34
35 int main(){
36
37
        string S;
38
       cin>>S;
39
        int n = S.size();
40
        int A[n];
41
        for(int i=0;i<n;i++)</pre>
42
            A[i] = S[i] - '0';
43
        Bit_Substrings(A,n);
44
45
46
        return 0;
47
```

Share code to others

Test details ▲

Test 1

Verdict: ACCEPTED

	input	
0000000000000		0

										(CO	rı	·e	t output
91	0	0	0	0	0	0	0	0	0	0	0	0	0	0 b

	user output	
91 0 0 0 0 0 0 0 0 0 0 0	000	©

Test 2

Verdict: ACCEPTED

	input	
1111111111111		()

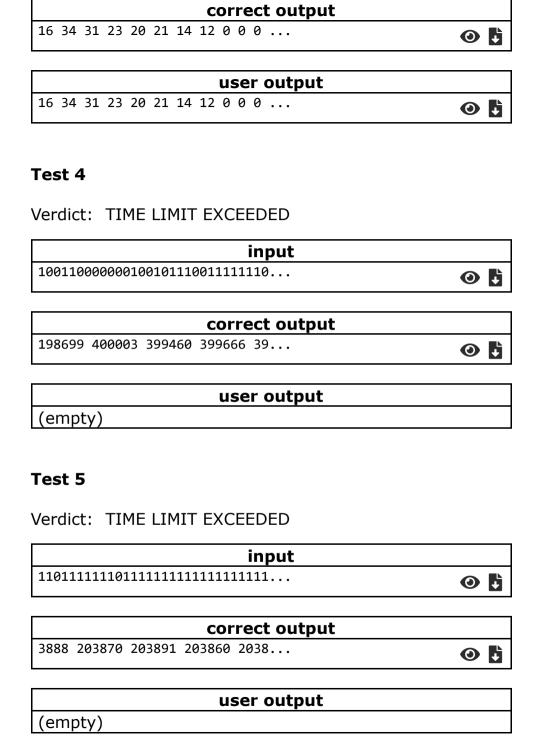
correct output	
0 14 13 12 11 10 9 8 7 6 5 4 3	()

user output	
0 14 13 12 11 10 9 8 7 6 5 4 3	0

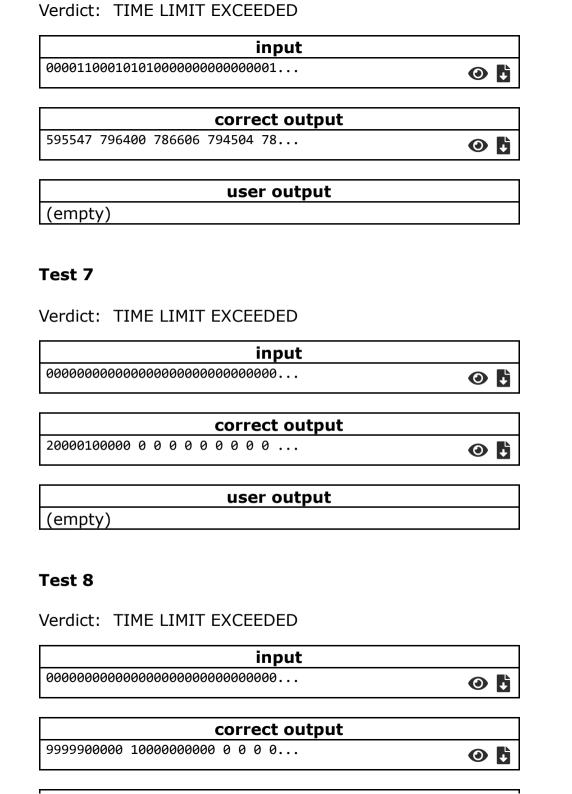
Test 3

Verdict: ACCEPTED

	input	
000101001011010100		O



Test 6



user output		
(empty)		