

Submission

ID	DATE	PROBLEM	STATUS	CPU	LANG
	TEST CASES				
8201948	17:50:54	Burrows-Wheeler	✓ Accepted	0.39 s	C++
	✓✓✓✓				

Submission contains 1 file: [download zip archive](#)

FILENAME	FILESIZE	SHA-1 SUM	
burrowswheeler.cpp	2548 bytes	522bc04b0fe18c6ff1b90ee6319cd8ebbde4775b	download

Edit and resubmit this submission.

burrowswheeler.cpp

```
1 #include <bits/stdc++.h>
2
3 using namespace std;
4
5 #define INF 999999
6 #define v vector
7 #define FOR(i,A,B,C) for(int i=A; i<B; i+=C)
8 #define FOR_bit(a,A,B,C) for(int a=A; a<B; a<=C)
9
10 v<int> arr;
11     ufijos;
12     .CP;
13 string T;
```

```

14 int n;
15 string Total;
16
17 void redimensionar(){
18     T = Total;
19     n = T.size();
20
21     sufijos.resize(n);
22     iota(sufijos.begin(), sufijos.end(), 0);
23     arr.resize(n);
24 }
25
26
27
28
29 void contar(int tam) {
30     int maximo = max(300, n);
31     v<int> aux(maximo, 0);
32     FOR(i,0,n,1){
33         ++aux[i + tam < n ? arr[i + tam] : 0];
34     }
35
36     int sum = 0;
37     FOR(i,0,maximo,1){
38         int t = aux[i];
39         aux[i] = sum;
40         sum += t;
41     }
42
43     v<int> tempsufijos(n);
44
45     FOR(i,0,n,1)
46         tempsufijos[aux[sufijos[i] + tam < n ? arr[sufijos[i] + tam] : 0]++] = sufijos[i];
47
48     swap(sufijos, tempsufijos);
49 }
50
51 void contAux(){
52     FOR_bit(k,1,n,1) {
53         contar(k);
54         contar(0);
55         Help vector<int> temparr(n);
56         int r = 0;
57         temparr[sufijos[0]] = r;

```

```

58     FOR(i,1,n,1){
59
60         int tmp = ((arr[sufijos[i]] == arr[sufijos[i - 1]]) && (arr[sufijos[i] + k] == arr[sufijos[i - 1] + k]));
61
62         temparr[sufijos[i]] = tmp? r: ++r;
63     }
64
65     swap(arr, temparr);
66     if (arr[sufijos[n - 1]] == n - 1)
67         break;
68 }
69 }
70
71 void sol() {
72
73     redimensionar();
74
75     FOR(i,0,n,1)
76         arr[i] = T[i];
77
78     contAux();
79 }
80
81
82 void construirLCP() {
83     vector<int> Phi(n);
84     vector<int> PLCP(n);
85     PLCP.resize(n);
86     Phi[sufijos[0]] = -1;
87     for (int i = 1; i < n; ++i)
88         Phi[sufijos[i]] = sufijos[i - 1];
89     for (int i = 0, L = 0; i < n; ++i) {
90         if (Phi[i] == -1) {
91             PLCP[i] = 0;
92             continue;
93         }
94         while ((i + L < n) && (Phi[i] + L < n) && (T[i + L] == T[Phi[i] + L]))
95             ++L;
96         PLCP[i] = L;
97         L = max(L - 1, 0);
98
99         Help resize(n);
100     for (int i = 0; i < n; ++i)
101         LCP[i] = PLCP[sufijos[i]];

```

```
102 }
103
104
105 int main() {
106
107     while (getline(cin, Total)) {
108         int n = Total.size();
109         Total += Total;
110         Total.pop_back();
111
112         Total += char(9);
113
114         sol();
115         constuirLCP();
116
117         for (auto i : sufijos) {
118             if (i >= n)
119                 continue;
120             cout << Total[i + n - 1];
121         }
122         cout << endl;
123     }
124
125     return 0;
126 }
```