CSES Problem Set

Substring Order I

Submission details

Task:	Substring Order I
Sender:	Rodry
Submission time:	2021-12-09 21:46:39
Language:	C++17
Status:	READY
Result:	ACCEPTED

Test results

test	verdict	time	
#1	ACCEPTED	0.01 s	»
#2	ACCEPTED	0.01 s	»
#3	ACCEPTED	0.12 s	»
#4	ACCEPTED	0.12 s	»
#5	ACCEPTED	0.06 s	»
#6	ACCEPTED	0.05 s	»
#7	ACCEPTED	0.14 s	»
#8	ACCEPTED	0.05 s	»
#9	ACCEPTED	0.12 s	»

Code

```
1 #include<bits/stdc++.h>
2 using namespace std;
3
4 #define INF 9999999
```


String Algorithms

...
Counting Patterns -
Pattern Positions -
Distinct Substrings -
Repeating Substring -
String Functions -
Substring Order I ✓
Substring Order II ×
Substring Distribution -

Your submissions

2021-12-09 21:46:39 ✓
2021-12-09 21:39:54 ×
2021-12-09 21:38:05 ×

```cpp

long long sa[INF], pos[INF], tmp[INF], lcp[INF];
long long  gap, N;
string S;

bool comp(long long  x, long long  y) {
    if (pos[x] != pos[y])
        return pos[x] < pos[y];
    x += gap;
    y += gap;
    return (x < N && y < N) ? pos[x] < pos[y] : x > y;
}

void sufijos() {
    for (int i = 0; i < N; i++)
        sa[i] = i, pos[i] = S[i];

    for (gap = 1;; gap <<= 1) {
        sort(sa, sa+N, comp);
        for (int i = 0; i < N-1; i++)
            tmp[i+1] = tmp[i] + comp(sa[i], sa[i+1]);
        for (int i = 0; i < N; i++)
            pos[sa[i]] = tmp[i];
        if (tmp[N - 1] == N - 1)
            break;
    }
}

void construirLCP() {
    for (int i = 0, k = 0; i < N; i++)
                if (pos[i] != N-1) {
        long long j = sa[pos[i] + 1];
        while (S[i + k] == S[j + k])
            k++;
        lcp[pos[i]] = k;
        if (k)
                        k--;
    }
}

int  main(){


    cin>>S;
        N = S.size();

```

```
51          sufijos();
52
53      construirLCP();
54
55      long long  k;
56          cin>>k;
57
58          long long  prev = 0;
59      long long  act = 0;
60
61
62      for (int i = 0; i < N; i++) {
63          if (act + (N-sa[i]) - prev >= k) {
64              long long  j = prev;
65              string ans = S.substr(sa[i], j);
66              while (act < k) {
67                  ans += S[sa[i]+j];
68                  act++;
69                  j++;
70              }
71              cout<<ans;
72              return 0;
73          }
74          act += (N-sa[i]) - prev;
75          prev = lcp[i];
76      }
77  }
```

Share code to others

Test details ▲

Test 1

Verdict: ACCEPTED

input
ababbaaabbabaabaabbaababa 10

correct output
aaabbabaab

user output
aaabbabaab

Test 2

Verdict: ACCEPTED

input
rmnxvouggsdgespsltsldcvkxtg 33

correct output
esps

user output
esps

Test 3

Verdict: ACCEPTED

input
aaaaaaaaaaaaaaaaaaaaaaaaaaaaa...

correct output
aaaaaaaaaaaaaaaaaaaaaaaaaaaaa...

user output
aaaaaaaaaaaaaaaaaaaaaaaaaaaaa...

Test 4

Verdict: ACCEPTED

input
abababababababababababababab...

correct output
bababababababababababababaaba...

user output
bababababababababababababaaba...

Test 5

Verdict: ACCEPTED

input
ababbbbaaaabaabbabaaabbaaaaaba...

correct output
aabbaababaaabbbaaaaaababaaaaba...

user output
aabbaababaaabbbaaaaaababaaaaba...

Test 6

Verdict: ACCEPTED

input
bslzdzbpuyxvovpeqxjhpnexwdheng...

correct output
fdzadhalzyzjstzcplofwhrvgshymp...

user output
fdzadhalzyzjstzcplofwhrvgshymp...

Test 7

Verdict: ACCEPTED

input
iybvzbbtkfqevdtjwqhezljzdkkjwi...

correct output
brdcxlfbsneugpmevkwmehndrzncoh...

user output
brdcxlfbsneugpmevkwmehndrzncoh...

Test 8

Verdict: ACCEPTED

input
gawaxmbhgoatjtxywopqckecliivyd...

correct output
phtjwgbtgbhslxxtprgbyppsnekyoy...

user output
phtjwgbtgbhslxxtprgbyppsnekyoy...

Test 9

Verdict: ACCEPTED

input
aaaaaaaaaaaaaaaaaaaaaaaaaaaaa...

correct output
aaaaaaaaaaaaaaaaaaaaaaaaaaaaa...

user output
aaaaaaaaaaaaaaaaaaaaaaaaaaaaa...