Submission

ID	DATE	PROBLEM	STATUS		CPU	LANG
	TEST CASES					
8192204	02:03:49	Dvaput	✔ **Accepted**		0.27 s	C++
	✔✔✔✔✔✔✔✔✔✔✔					

Submission contains 1 file: [download zip archive]

FILENAME	FILESIZE	SHA-1 SUM	
dvaput.cpp	1433 bytes	c338f34867e64c7a8d85539a6b84368eb8727119	[download]

Edit and resubmit this submission.

dvaput.cpp

```
 1  #include <bits/stdc++.h>
 2  using namespace std;
 3
 4  const int maxsize = 2000000;
 5  vector<int> _suffix_arr,_pos;
 6  int N,indx;
 7  string S;
 8
 9  bool comp(int x, int y){
10      if (_pos[x] != _pos[y])
11          return _pos[x] < _pos[y];
12      x+=indx;
13      y+=indx;
```

```
14        return (x<N &&y<N)?_pos[x]<_pos[y]:x>y;
15  }
16
17  void suffix_array(){
18      for (int i = 0; i < N; i++){
19          _suffix_arr.push_back(i);
20          _pos.push_back(S[i]);
21      }
22      int tmp[maxsize];
23      for (indx = 1;; indx <<= 1) {
24          sort(_suffix_arr.begin(), _suffix_arr.end(), comp);
25          for (int i = 0; i < N-1; i++)
26              tmp[i+1] = tmp[i] + comp(_suffix_arr[i], _suffix_arr[i+1]);
27          for (int i = 0; i < N; i++)
28              _pos[_suffix_arr[i]] = tmp[i];
29          if (tmp[N - 1] == N - 1)
30              break;
31      }
32  }
33
34  int lcp(){
35      int longest_prefix_size=0;
36      vector<int> _lcp(N);
37      for (int i=0,k=0; i<N;i++){
38          if (_pos[i] != N-1){
39              int j = _suffix_arr[_pos[i] + 1];
40              while (S[i + k] == S[j + k])
41                  k++;
42              _lcp[_pos[i]+1] = k;
43              longest_prefix_size = max(longest_prefix_size,_lcp[_pos[i]+1]);
44              if (k) k--;
45          }
46      }
47      return longest_prefix_size;
48  }
49
50  int Dvaput(){
51      suffix_array();
52      return lcp();
53  }
54
55      Help    (){
56      ios_base::sync_with_stdio(false);
57      cin.tie(NULL);
```

```
58    cin>>N>>S;
59    cout<<Dvaput();
60    return 0;
61  }
```