IUST - 8 silandr - Notebook

Contents

1	Template - SublimeConfig	1
	1.1 Sublime Config	
	1.2 Template	1
2	Implementation	1
	2.1 Token	1
3	String	1
	3.1 Z-Function	1
4	Graph	2
	4.1 2sat	2

1 Template - SublimeConfig

1.1 Sublime Config

```
"shell_cmd": "g++ ${file_name} && gnome-terminal -- bash -c './a.out; read'"
```

1.2 Template

```
#pragma GCC optimize ("Ofast, unroll-loops, fast-math")
#include < bits / stdc++.h>
using namespace std;
typedef long long ll;
#define pll pair<11 , 11 >
\#define all(x) (x).begin(),(x).end()
\#define SZ(x) (ll)(x).size()
#define X first
#define Y second
#define mp make pair
#define pii pair<int , int>
#define vec vector
#define file io freopen("input.txt", "r", stdin); freopen("output.txt",
     "w", stdout);
#define migmig ios::sync with stdio(false); cin.tie(0); cout.tie(0);
#define pb push back
#define ld long double
// BIG p : 100000000000037 , 100000000003
'll poww(ll a, ll b, ll md) {
    return (!b?1: (b&1?a*poww(a*a%md, b/2, md)%md:
        poww(a * a % md, b / 2, md) % md));
const int \max = 1000*100+5;
const 11 \text{ inf} = 9223372036854775807;
const 11 \mod = 1e9 + 7;
const int lg = 20;
```

2 Implementation

2.1 Token

```
vector<string> token(string& s, string& del)
{
    // string inp = "Hello,World;This|is.GeeksForGeeks pam";
    // string del = ",;,| ";
    // vector<string> ans = token(inp, del);
    vector<string> ans;
    int start = 0, last = 0;
    while ((last = s.find_first_of(del, start)) != string::npos)
        {
        if (last != start)
            ans.pb(s.substr(start, last - start));
        start = last + 1;
    }
    if (start != s.length())
        ans.pb(s.substr(start));
    return ans;
}
```

3 String

3.1 Z-Function

```
const 11 \max = 2e5 + 10;
11 z [maxn];
// Pattern matching: maximum character matching of start at index i
// text && pattern ----> s = pattern + '$' + text
// z[i] = number of match prifix start index i
void z_function(string s)
    ll left = 0, right = 0;
    for (int i = 1; i < s. size(); i++)
        if(i \le right \&\& z[i - left] < right - i + 1)
            z[i] = z[i - left];
        else
            if(i \le right)
                left = i;
                left = right = i;
            while (right < s. size () && s[right] == s[right - left])
                right++;
            z[i] = right - left;
            right --;
```

}

4 Graph

4.1 2sat

```
//harki mesle i khodesh 2 * i notesh 2 * i + 1 he
//yale u -> v yani agar u yek baseh v ham bayad yek bashe
vector<int> out[maxn * 2] , in[maxn * 2] , topol , adj[maxn];

bool visited[maxn * 2];

int color[maxn * 2] , c , val[maxn] , r[maxn] ;

void add_edge(int v , int u)
{
        out[v].pb(u);
        in[u].pb(v);
}

void dfs(int v)
{
        visited[v] = 1;
```

```
for (auto u : out [v])
               if (! visited [u])
                       dfs(u);
       topol.pb(v);
void sfd(int v)
       visited[v] = 1;
       color[v] = c;
       for (auto u : in [v])
               if (!visited[u])
                       sfd(u);
for (int i = 0; i < 2 * n; i++)
               if (! visited [i])
                       dfs(i);
reverse(topol.begin() , topol.end());
memset(visited , 0 , size of visited);
if (! visited [v])
sfd (v) , c++;
       for (int i = 1; i \le m; i++)
               if(color[2 * i] = color[2 * i + 1])
                       return cout << "NO" << endl , 0;
```