Competitive Programming Notebook

Programadores Roblox

Contents

1 DP

2 String

3 Geometry

4 Graph

4.1 Dijkstra

```
// Caminho mÃŋnimo com pesos positivos.
// Complexidade: O((V + E) log V).
vector<int> dijkstra(int S) {
    vector<bool> vis(MAXN, 0);
    vector<ll> dist(MAXN, LLONG_MAX);
    dist[S] = 0;
    priority_queue<pii, vector<pii>, greater<pii>> pq;
    pq.push({0, S});
    while(pq.size()) {
        ll v = pq.top().second;
        pq.pop();
        if(vis[v]) continue;
        vis[v] = 1;
        for(auto &[peso, vizinho] : adj[v]) {
            if(dist[vizinho] > dist[v] + peso) {
                dist[vizinho] = dist[v] + peso;
                pq.push({dist[vizinho], vizinho});
            }
        }
    }
    return dist;
}
```

5 Math

5.1 Fexp

```
// a^e mod m
// O(log n)

ll fexp(ll a, ll e, ll m) {
    a %= m;
    ll ans = 1;
    while (e > 0){
        if (e & 1) ans = ansa % m;
        a = aa % m;
        e /= 2;
    }
    return ans%m;
}
```

6 DS

7 Primitives

8 General