

Олимпиада СПбГУ по информатике 2018/19 учебного года

Шарипов Дениль Ильшатович

A	B	C	D	E	F	Sum
100	100	100	60	100	100	560

Task A (100)

```
#include <bits/stdc++.h>
#define f first
#define s second

using namespace std;

typedef long long ll;
typedef pair <int, int> pii;

int main()
{
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    int n, m;
    cin >> n >> m;
    while(n < m)n *= 2;
    if(n == m)cout << "Yes\n";
    else cout << "No\n";
    return 0;
}
```

Task B (100)

```
#include <bits/stdc++.h>
#define f first
#define s second

using namespace std;

typedef long long ll;
typedef pair <int, int> pii;

int main()
{
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    int n;
    cin >> n;
    string s;
    cin >> s;
    bool check = 0;
    for(int i = 0; i < n - 1; i++){
        for(int j = i; j < min(i + 10, n - 1); j++){
            if(s[j] == 'o' && s[j + 1] == 'r')
                check = 1;
            swap(s[i], s[i + 1]);
            for(int j = i; j < min(i + 10, n - 1); j++){
                if(s[j] == 'o' && s[j + 1] == 'r')
                    check = 1;
            }
            swap(s[i], s[i + 1]);
        }
        if(check) cout << "Yes\n";
        else cout << "No\n";
        return 0;
    }
}
```

Task C (100)

```
#include <bits/stdc++.h>
#define f first
#define s second

using namespace std;

typedef long long ll;
typedef pair <int, int> pii;

int n;
vector < vector <int> > g;
vector <int> ans;
vector <int> sz;

void dfs(int v, int p){
    for(int i = 0; i < g[v].size(); i++){
        int to = g[v][i];
        if(to == p) continue;
        dfs(to, v);
        sz[v] += sz[to];
    }

    int mx = 0;
    for(int i = 0; i < g[v].size(); i++){
        int to = g[v][i];
        if(to == p) mx = max(mx, n - sz[v] + 1);
        mx = max(mx, sz[to] + 1);
    }
    ans[v] = mx;
}

int main()
{
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    cin >> n;
    if(n == 1){
        cout << 1 << '\n';
        return 0;
    }
    g.resize(n), ans.resize(n), sz.resize(n, 1);
    for(int i = 0; i < n - 1; i++){
        int u, v;
        cin >> u >> v;
        u--, v--;
        g[u].push_back(v);
        g[v].push_back(u);
    }
    dfs(0, -1);
    for(auto x: ans) cout << x << ' ';
    return 0;
}
```

Task D (60)

```
#include <bits/stdc++.h>

using namespace std;

int main()
{
    string type;
    cin >> type;
    if(type == "split"){
        int t, n, p;
        cin >> t >> n >> p;
        if(n == 3 && p == 7){
            while(t--){
                string pass;
                cin >> pass;
                string pref = pass.substr(0, 6) + "a";
                string suf = pass.substr(3, 6) + "b";
                string mid = pass.substr(0, 3) + pass.substr(6, 3) + "c";
                cout << pref << '␣' << suf << '␣' << mid << '\n';
            }
        }
        else{
            while(t--){
                string pass;
                cin >> pass;
                string pref = pass.substr(0, 6) + "a";
                string suf = pass.substr(3, 6) + "b";
                string mid = pass.substr(0, 3) + pass.substr(6, 3) + "c";
                cout << pref << '␣' << suf << '␣' << mid << '␣' << pref << '␣' << suf << '\n';
            }
        }
    }
    else{
        int t, n, p;
        cin >> t >> n >> p;
        if(n == 3 && p == 7){
            while(t--){
                string pref, suf;
                cin >> pref >> suf;
                if(suf[6] < pref[6])swap(suf, pref);
                if(pref[6] == 'a' && suf[6] == 'b')
                    cout << pref.substr(0, 6) << suf.substr(3, 3) << '\n';
                else if(pref[6] == 'a' && suf[6] == 'c')
                    cout << pref.substr(0, 6) << suf.substr(3, 3) << '\n';
                else
                    cout << suf.substr(0, 3) + pref.substr(0, 6) << '\n';
            }
        }
        else{
            while(t--){
                string pref, suf, mid;
                cin >> pref >> suf >> mid;
                vector<int> cnt(3);
                if(suf[6] < pref[6])swap(pref, suf);
                if(mid[6] < pref[6])swap(mid, pref);
                if(suf[6] > mid[6])swap(suf, mid);
                cnt[pref[6] - 'a']++;
                cnt[suf[6] - 'a']++;
                cnt[mid[6] - 'a']++;
                if(cnt[0] && cnt[1])
                    cout << pref.substr(0, 6) << mid.substr(3, 3) << '\n';
                else if(cnt[0] && cnt[2])
                    cout << pref.substr(0, 6) << mid.substr(3, 3) << '\n';
                else
                    cout << mid.substr(0, 3) + suf.substr(0, 6) << '\n';
            }
        }
    }
}
```

```
}    return 0;
```

Task E (100)

```
def len(x1, y1, x2, y2):
    return ((x1 - x2) * (x1 - x2) + (y1 - y2) * (y1 - y2))

n = int(input())
xs = []
ys = []
for i in range(n):
    x, y = map(int, input().split())
    xs.append(x)
    ys.append(y)
px, py = map(int, input().split())
qx, qy = map(int, input().split())

l = len(px, py, qx, qy)
X = px + (qx - px) * (10 ** 20)
Y = py + (qy - py) * (10 ** 20)
num = 0
check = 0
for i in range(n):
    if len(xs[i], ys[i], X, Y) < len(xs[num], ys[num], X, Y):
        num = i

for i in range(n):
    if len(xs[i], ys[i], X, Y) == len(xs[num], ys[num], X, Y):
        check += 1

if check == 1:
    print(num + 1)
else:
    print(-1)
```

Task F (100)

```
#include <bits/stdc++.h>
#define f first
#define s second
#define int long long

using namespace std;

typedef long long ll;
typedef pair <int, int> pii;

int dp[111111][2];
int cnt[111111][2][2]; // type - product

signed main()
{
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    int n, k;
    cin >> n >> k;
    vector < vector <int> > p(n, vector <int> (2));
    for(int i = 0; i < n; i++){
        int ri, bi;
        cin >> ri >> bi;
        p[i][0] = ri, p[i][1] = bi;
    }

    // if(n <= 30 && k <= 30){
    //     int mx = (1 << n);
    //     int ans = 0;
    //     for(int i = 0; i < mx; i++){
    //         int red = 0, blue = 0, cur_ans = 0;
    //         for(int j = 0; j < n; j++){
    //             int bit = ((i >> j) & 1);
    //             if(bit == 0)
    //                 red += k;
    //             else
    //                 blue += k;
    //             cur_ans += min(red, p[j][0]);
    //             cur_ans += min(blue, p[j][1]);
    //             red = max(red - p[j][0], 0ll);
    //             blue = max(blue - p[j][1], 0ll);
    //         }
    //         ans = max(ans, cur_ans);
    //     }
    //     cout << ans << '\n';
    //     return 0;
    // }
    cnt[0][0][0] = max(k - p[0][0], 0ll);
    cnt[0][1][1] = max(k - p[0][1], 0ll);
    dp[0][0] = min(k, p[0][0]);
    dp[0][1] = min(k, p[0][1]);

    for(int i = 1; i < n; i++){
        for(int j = 0; j < 2; j++){
            for(int j1 = 0; j1 < 2; j1++){
                if(n > 30 && k > 30){
                    if(dp[i - 1][j1] + min(cnt[i - 1][j][j1] + k, p[i][j]) + min(cnt[i - 1][j ^
                        1][j1], p[i][j ^ 1]) >= dp[i][j]){
                        dp[i][j] = dp[i - 1][j1] + min(cnt[i - 1][j][j1] + k, p[i][j]) + min(cnt[i
                            - 1][j ^ 1][j1], p[i][j ^ 1]);
                        cnt[i][j][j] = max(cnt[i - 1][j][j1] + k - p[i][j], 0ll);
                        cnt[i][j ^ 1][j] = max(cnt[i - 1][j ^ 1][j1] - p[i][j ^ 1], 0ll);
                    }
                }
                else{
                    if(dp[i - 1][j1] + min(cnt[i - 1][j][j1] + k, p[i][j]) + min(cnt[i - 1][j ^
                        1][j1], p[i][j ^ 1]) >= dp[i][j]){
                        dp[i][j] = dp[i - 1][j1] + min(cnt[i - 1][j][j1] + k, p[i][j]) + min(cnt[i
                            - 1][j ^ 1][j1], p[i][j ^ 1]);
                        cnt[i][j][j] = max(cnt[i - 1][j][j1] + k - p[i][j], 0ll);
                        cnt[i][j ^ 1][j] = max(cnt[i - 1][j ^ 1][j1] - p[i][j ^ 1], 0ll);
                    }
                }
            }
        }
    }
}
```

```

        1][j1], p[i][j ^ 1]) > dp[i][j]){
        dp[i][j] = dp[i - 1][j1] + min(cnt[i - 1][j][j1] + k, p[i][j]) + min(cnt[i
            - 1][j ^ 1][j1], p[i][j ^ 1]);
        cnt[i][j][j] = max(cnt[i - 1][j][j1] + k - p[i][j], 011);
        cnt[i][j ^ 1][j] = max(cnt[i - 1][j ^ 1][j1] - p[i][j ^ 1], 011);
    }
}
}
cout << max(dp[n - 1][0], dp[n - 1][1]) << '\n';
return 0;
}

```