Олимпиада СПбГУ по информатике 2018/19 учебного года

Бородин Евгений Сергеевич

A	B	C	D	E	F	Sum
100	100	60	60	35	0	355

Task A (100)

```
#include <iostream>
#include <vector>
#include <math.h>
#include <algorithm>

using namespace std;

int main(){
    int n,m;
    cin >> n >> m;
    while (m > n) {
        n*=2;
    }
    if (n == m) {
        cout << "Yes";
        return 0;
    }
    if (n > m) {
        cout << "No";
        return 0;
    }

    cout << 123456;
    return 0;
}
```

Task B (100)

```cpp
#include <iostream>
#include <vector>
#include <math.h>
#include <algorithm>

using namespace std;

int main(){
    int n;
    cin >> n;
    string a;
    cin >> a;
    for (int i = 0; i < n-1; i++) {
        if (a[i] == 'o' && a[i+1] == 'r') {
            cout << "Yes";
            return 0;
        }
        if (a[i] == 'r' && a[i+1] == 'o') {
            cout << "Yes";
            return 0;
        }
        if (a[i] == 'o' && i!=n-2) {
            if (a[i+2] == 'r') {
                cout << "Yes";
                return 0;
            }
        }
    }
    cout << "No";
    return 0;
}
```

Task C (60)

```
#include <iostream>
#include <vector>
#include <math.h>
#include <algorithm>

using namespace std;
vector <vector <int> > a;
vector <int> answer;
vector <bool> used;
vector <int> now;
int count1;

int dfs(int q) {
    count1++;
    used[q] = true;
    for (int i = 0; i < a[q].size(); i++) {
        if (!used[a[q][i]]) {
            dfs(a[q][i]);
        }
    }
    return 0;
}

int main(){
    int n;
    cin >> n;
    answer.resize(n,1);
    a.resize(n);

    for (int i = 0; i < n-1; i++) {
        int q,w;
        cin >> q >> w;
        a[w-1].push_back(q-1);
        a[q-1].push_back(w-1);
    }
    if (n > 10000) {
        for (int i = 0; i < n;i++) {
            cout << max(1+i,n-i) << "␣";
        }
        return 0;
    }
    /*for (int i =0 ; i < n; i++) {
        for (int j = 0; j < a[i].size(); j++) {
            cout << a[i][j] << " ";
        }
        cout << endl;
    }*/
    used.resize(n);
    for (int j = 0; j < n; j++) {
    int max1 = 0;
    for (int i = 0; i < n; i++) {
        used[i] = false;
    }
    used[j] = true;
    for (int i = 0; i < a[j].size(); i++) {
        count1 = 0;
        if (!used[a[j][i]]) {
            dfs(a[j][i]);
        }
        if (count1 > max1) {
            max1 = count1;
        }
    }
    answer[j] = max1+1;
    }
    for (int i = 0; i < n; i++) {
        cout << answer[i] << "␣";
    }
    return 0;
}
```

Task D (60)

```
#include <iostream>
#include <vector>
#include <math.h>
#include <algorithm>

using namespace std;
int n,p;

int split() {
    string q;
    cin >> q;
    int index = 0;
    if (n == 3 || n == 5) {
        for (int ind = 0; ind < n; ind++) {
            cout << (char)('a' + ind%3);
            for (int i = 0; i < p-1; i++) {
                if (index == q.size()) {
                    index=0;
                }
                cout << q[index];
                index++;
            }
            cout << " ";
        }
        cout << endl;
    }
    return 0;
}

int merge1(){
    if (n==3) {
        string a,b;
        cin >> a >> b;
        if (a[0] == 'a') {
            for (int i = 1; i < 7; i++) {
                cout << a[i];
            }
            if (b[0] == 'b'){
                cout << b[1] << b[2] << b[3];
            }
            else {
                cout << b[4] << b[5] << b[6];
            }
        }
        else if (a[0] == 'b') {
            for (int i = 4; i < 7; i++) {
                cout << a[i];
            }
            if (b[0] == 'a'){
                for (int i = 4; i < 7; i++) {
                    cout << b[i];
                }
                for (int i = 1; i < 4; i++) {
                    cout << a[i];
                }
            }
            else {
                for (int i = 1; i < 7; i++) {
                    cout << b[i];
                }
            }
        }
        else{
            if (b[0] == 'a') {
                cout << b[1] << b[2] << b[3];
                for (int i = 1; i < 7; i++) {
                    cout << a[i];
                }
            }
            else {
                cout << b[4] << b[5] << b[6];
                for (int i = 1; i < 7; i++) {
                    cout << a[i];
                }
            }
        }
        cout << endl;
    }


    else if (n==5) {
        string a,b,c;
        cin >> a >> b >> c;
        if (a == b) {
            b = c;
        }
        if (a[0] == 'a') {
            for (int i = 1; i < 7; i++) {
                cout << a[i];
            }
            if (b[0] == 'b'){
                cout << b[1] << b[2] << b[3];
            }
            else {
                cout << b[4] << b[5] << b[6];
            }
        }
        else if (a[0] == 'b') {
            for (int i = 4; i < 7; i++) {
                cout << a[i];
            }
            if (b[0] == 'a'){
                for (int i = 4; i < 7; i++) {
                    cout << b[i];
                }
                for (int i = 1; i < 4; i++) {
                    cout << a[i];
                }
            }
            else {
                for (int i = 1; i < 7; i++) {
                    cout << b[i];
                }
            }
        }
        else{
            if (b[0] == 'a') {
                cout << b[1] << b[2] << b[3];
                for (int i = 1; i < 7; i++) {
                    cout << a[i];
                }
            }
            else {
                cout << b[4] << b[5] << b[6];
                for (int i = 1; i < 7; i++) {
                    cout << a[i];
                }
            }
        }
        cout << endl;
    }
    return 0;
}

int main(){
    string s;
    cin >> s;
    if (s == "split") {
        int t;
        cin >> t >> n >> p;
        for (int i  = 0; i < t; i++) {
            split();
        }
    }
    else {
        int t;
        cin >> t >> n >> p;
        for (int i  = 0; i < t; i++) {
            merge1();
        }
    }
    return 0;
}
```

Task E (35)

```cpp
#include <iostream>
#include <vector>
#include <math.h>
#include <algorithm>

using namespace std;

int main(){
    int n;
    vector <pair <long long,long long> > a;
    cin >> n;
    a.resize(n);
    int ans = 1;
    for (int i = 0; i < n; i++){
        long long q,w;
        cin >> q >> w;
        a[i] = make_pair(q,w);
    }
    long long xp,xq,yp,yq;
    bool t = false;
    cin >> xp >> yp >> xq >> yq;
    if (xp < xq) {
        long long min1 = abs(a[0].second), max1 = a[0].first;
        for (int i = 1; i < n; i++) {
            if (a[i].first > max1) {
                max1 = a[i].first;
                min1 = abs(a[i].second);
                t = false;
                ans = i+1;
            }
            else if (a[i].first == max1) {
                if (min1 > abs(a[i].second)) {
                    min1 = abs(a[i].second);
                    ans = i+1;
                }
                else if (min1 == abs(a[i].second)) {
                    t = true;
                }
            }
        }
    }
    else {
        long long min1 = abs(a[0].second), max1 = a[0].first;
        for (int i = 1; i < n; i++) {
            if (a[i].first < max1) {
                max1 = a[i].first;
                min1 = abs(a[i].second);
                t = false;
                ans = i+1;
            }
            else if (a[i].first == max1) {
                if (min1 > abs(a[i].second)) {
                    min1 = abs(a[i].second);
                    ans = i+1;
                }
                else if (min1 == abs(a[i].second)) {
                    t = true;
                }
            }
        }
    }

    if (t) {
        cout << -1;
    }
    else {
        cout << ans;
    }
    return 0;
}
```

Task F (0)

```cpp
#include <iostream>
#include <vector>
#include <math.h>
#include <algorithm>

using namespace std;

int main(){
    int n,k;
    vector <pair <long long,long long> > a;
    vector <long long> b;
    int index = 1;
    a.push_back(make_pair(0,0));
    a.push_back(make_pair(0,0));
    b.push_back(0);
    b.push_back(0);
    cin >> n >> k;
    for (int i = 0; i < n; i++){
        long long q,w;
        cin >> q >> w;
        for (int j = index; j < index*2;j++){
            a.push_back(make_pair(k-q+a[j].first,a[j].second-w));
            a.push_back(make_pair(a[j].first-q,k-w+a[j].second));
            long long p = b[j];
            if (k-q+a[j].first < 0) {
                p+=abs(k-q+a[j].first);
            }
            if (a[j].second-w < 0) {
                p+=abs(a[j].second-w);
            }
            b.push_back(p);
            p = b[j];
            if (a[j].first-q < 0) {
                p+=abs(a[j].first-q);
            }
            if (k-w+a[j].second < 0) {
                p+=abs(k-w+a[j].second);
            }
            b.push_back(p);
        }
        index*=2;
    }
    long long max1 = -1;

    for (int i = index; i < index*2; i++) {
        long long p = 0;
        if (a[i].first < 0) {
            p+=abs(a[i].first);
        }
        if (a[i].second < 0) {
            p+=abs(a[i].second);
        }
        if (p > max1) {
            max1 = p;
        }
    }
    cout << max1;
    return 0;
}
```