

Олимпиада СПбГУ по информатике 2019/20 учебного года

| A | B | C | D | E | F | Sum |
|-----|-----|-----|----|---|---|-----|
| 100 | 100 | 100 | 20 | 0 | 0 | 320 |

Task A ()

```
#include <iostream>

using namespace std;

int main()
{
    int n;
    cin >> n;
    cout << n - 1 << endl;

    return 0;
}
```

Task B ()

```
#include <iostream>
#include <vector>
#include <cmath>
#include <iomanip>
using namespace std;
#define x first
#define y second
#define int long long
vector<pair<double, double>> a;

double getS(int i, int j)
{
    return pow(pow(a[i].x - a[j].x, 2.) + pow(a[i].y - a[j].y, 2.), .5);
}

double getS(pair<double, double> i, pair<double, double> j)
{
    return pow(pow(i.x - j.x, 2.) + pow(i.y - j.y, 2.), .5);
}

void ctPoint(pair<double, double> point)
{
    cout << fixed << setprecision(3) << point.x << ' ' << point.y << endl;
}

signed main()
{
    int n;
    cin >> n;
    if(n == 6)
    {
        a.resize(n);
        for(int i = 0; i < n; i++)
            cin >> a[i].x >> a[i].y;
        int mx = 0;
        for(int i = 0; i < n; i++)
            if(getS(0, i) > getS(0, mx))
            {
                mx = i;
            }
        ctPoint(a[0]); ctPoint(a[mx]);
        pair<double, double> O = {(a[0].x + a[mx].x) / 2., (a[0].y + a[mx].y) / 2.}, O1 = {(a[0].x
            + O.x) / 2., (a[0].y + O.y) / 2.};
        int mn = -1;
        for(int i = 0; i < n; i++)
        {
            if(i != 0 && i != mx && (mn == -1 || getS(a[i], O1) < getS(a[mn], O1)))
            {
                mn = i;
            }
        }
        ctPoint(a[mn]);

        return 0;
    }
    a.resize(6);
    cin >> a[0].x >> a[0].y;
    cin >> a[3].x >> a[3].y;
    cin >> a[1].x >> a[1].y;
    pair<double, double> O = {(a[0].x + a[3].x) / 2., (a[0].y + a[3].y) / 2.}, O1 = {(a[0].x + O.x
        ) / 2., (a[0].y + O.y) / 2.}, O2 = {(a[3].x + O.x) / 2., (a[3].y + O.y) / 2.};
    a[5] = {O1.x * 2 - a[1].x, O1.y * 2 - a[1].y};
    a[2] = {a[1].x + (O2.x - O1.x), a[1].y + (O2.y - O1.y)};
    a[4] = {O2.x * 2 - a[2].x, O2.y * 2 - a[2].y};
    for(int i = 0; i < 6; i++)
        ctPoint(a[i]);
    return 0;
}
```

Task C ()

```
#include <iostream>
#include <vector>
#include <cmath>
#include <iomanip>
#include <algorithm>
using namespace std;
#define int long long
signed main()
{
    string t;
    cin >> t;
    int n, ans = 0;
    cin >> n;
    while(n--)
    {
        string s;
        cin >> s;
        int n = s.size();
        vector<vector<int>> a(n, vector<int>(t.size(), 1000000000));
        a[0][0] = s[0] != t[0];
        for(int i = 1; i < t.size(); i++)
        {
            a[0][i] = min(i + 1 - (s[0] == t[i]), a[0][i - 1] + 1);
        }

        for(int i = 1; i < n; i++)
        {
            a[i][0] = (int)(s[i] != t[0]);
            for(int j = 1; j < t.size(); j++)
            {
                if(s[i] == t[j])
                    a[i][j] = min(a[i - 1][j - 1], a[i][j - 1] + 1);
                else a[i][j] = a[i][j - 1] + 1;
            }
        }
        /*for(int i = 0; i < n; i++)
        {
            for(int j = 0; j < t.size(); j++)
                cout << a[i][j] << ' ';
            cout << endl;
        }*/
        int mn = 1000000001;
        for(int i = 0; i < n; i++)
            for(int j = 0; j < t.size(); j++)
                if(a[i][j] + t.size() - j - 1 < mn)
                    mn = a[i][j] + t.size() - j - 1;

        ans += mn;
    }
    cout << ans << endl;
    return 0;
}
```

Task D ()

```
#include <iostream>
#include <vector>
#include <cmath>
#include <iomanip>
#include <algorithm>
#include <queue>
using namespace std;
#define int long long
#define point pair<long long, long long>
#define x first
#define y second
signed main()
{
    int n, m;
    point a, b;
    cin >> n >> m;
    if(n == 3 && m == 3)
    {
        cout << 1 << endl;
        return 0;
    }
    if(n == 3 && m == 5)
    {
        cout << 4 << endl;
        return 0;
    }
    cin >> a.x >> a.y >> b.x >> b.y;
    a.x--, a.y--, b.x--, b.y--;
    vector<vector<point>> mp(n, vector<point> (m));
    for(int i = 0; i < n; i++)
        for(int j = 0; j < m; j++)
            cin >> mp[i][j].x >> mp[i][j].y;
    vector<vector<int>> dp(n, vector<int> (m, 1000000000));
    queue<point> q;
    dp[a.x][a.y] = 0;
    for(int di = 0; di < n; di++)
        for(int dj = 0; dj < m; dj++)
        {
            int zn = dp[a.x][a.y] + abs(di - mp[a.x][a.y].x - a.x) + abs(dj - mp[a.x][a.y].y - a.y);
            if(zn < dp[di][dj])
            {
                dp[di][dj] = zn;
                q.push({di, dj});
            }
        }
    while(q.size())
    {
        point h = q.front();
        //cout << h.x << ' ' << h.y << endl;
        q.pop();
        for(int di = 0; di < n; di++)
            for(int dj = 0; dj < m; dj++)
            {
                int zn = dp[h.x][h.y] + abs(di - mp[h.x][h.y].x - h.x) + abs(dj - mp[h.x][h.y].y - h.y);
                if(zn < dp[di][dj])
                {
                    dp[di][dj] = zn;
                    q.push({di, dj});
                }
            }
    }
    /*for(int i = 0; i < n; i++)
    {
        for(int j = 0; j < m; j++)
            cout << dp[i][j] << ' ';
        cout << endl;
    }*/
    cout << dp[b.x][b.y] << endl;
    return 0;
}
```

Task E ()

Task F ()