

# Олимпиада СПбГУ по информатике 2019/20 учебного года

A	B	C	D	E	F	Sum
100	100	100	40	6	0	346

## Task A ()

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

#define return0 ios_base::sync_with_stdio(false);cin.tie(0);cout.tie(0);
#define se second
#define fi first

typedef long long ll;
typedef unsigned long long ull;

using namespace std;

int main()
{
    return0
    ll a;
    cin >> a;
    cout << a-1;

    return 0;
}
```

## Task B ()

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

#define return0 ios_base::sync_with_stdio(false); cin.tie(0); cout.tie(0);
#define se second
#define fi first

typedef long long ll;
typedef long double ld;
typedef unsigned long long ull;

using namespace std;

struct cord{
ld x;
ld y;
cord (ld a=-1, ld b=-1){
    x=a;
    y=b;
}
void out(){
    cout << x << " " << y << endl;
}
};

cord sumvect (cord a, cord b){
    return cord (a.x+b.x,a.y+b.y);
}

cord vect (cord a, cord b){
    return cord (b.x-a.x,b.y-a.y);
}

bool operator< (cord &a, cord &b){
    if (a.x==b.x){
        return a.y<b.y;
    }
    return a.x<b.x;
}

int main()
{
    return0
    cord a,b,c,d,e,f;
    int n;
    cin >> n;
    if (n==6){
        vector <cord> tab (6);
        for (int i=0; i<6; i++){
            cin >> tab[i].x >> tab[i].y;
        }
        sort (tab.begin(),tab.end());
        tab[0].out();
        int cnt=0;
        for (int i=1; i<6; i++){
            if (tab[i].y>=tab[0].y){
                tab[i].out();
                cnt++;
            }
            if (cnt==2){
                return 0;
            }
        }
        return 0;
    }
    if (n==3){
        cin >> a.x >> a.y >> b.x >> b.y >> c.x >> c.y;
        f=(vect(b,c));
        a.out();
        b.out();
        c.out();
        d=cord(a.x+2*f.x,a.y+2*f.y);
        d.out();
        e=vect(b,a);
    }
}
```

```
        e=cord(e.x+d.x,d.y+e.y);  
        e.out();  
        f=cord(e.x-f.x,e.y-f.y);  
        f.out();  
    }  
    return 0;  
}
```

## Task C ()

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

#define return0 ios_base::sync_with_stdio(false);cin.tie(0);cout.tie(0);
#define se second
#define fi first

typedef long long ll;
typedef unsigned long long ull;

using namespace std;

int main()
{
    return0
    string str;
    cin >> str;
    int n,answer=0;
    cin >> n;
    while (n--){
        string tmp;
        cin >> tmp;
        int lolans=1E9;
        for (int i=0; i<tmp.size(); i++){
            int kekw=i,ans=0;
            for (int j=0; j<str.size(); j++){
                if (kekw<tmp.size()){
                    if (tmp[kekw]==str[j]){
                        kekw++;
                    } else {
                        ans++;
                    }
                } else {
                    ans++;
                }
            }
            lolans=min(lolans,ans);
        }
        answer+=lolans;
    }
    cout << answer;

    return 0;
}
```

## Task D ()

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

#define return0 ios_base::sync_with_stdio(false);cin.tie(0);cout.tie(0);
#define se second
#define fi first
typedef long long ll;
typedef long double ld;
typedef unsigned long long ull;

using namespace std;
struct cord
{
    ll x;
    ll y;
    cord (ll a=-1, ll b=-1)
    {
        x=a;
        y=b;
    }
    void out()
    {
        cout << x << " " << y << endl;
    }
};

cord sumvect (cord a, cord b)
{
    return cord (a.x+b.x,a.y+b.y);
}

cord vect (cord a, cord b)
{
    return cord (b.x-a.x,b.y-a.y);
}

bool operator< (cord &a, cord &b)
{
    if (a.x==b.x)
    {
        return a.y<b.y;
    }
    return a.x<b.x;
}

using namespace std;

int main()
{
    return0
    int n,m;
    cin >> n >> m;

    cord h,t;
    cin >> t.x >> t.y >> h.x >> h.y;
    t.x--;
    t.y--;
    h.x--;
    h.y--;

    vector<vector<cord>> tab (n,vector<cord> (m));
    for (int i=0; i<n; i++)
    {
        for (int j=0; j<m; j++)
        {
            cin >> tab[i][j].x >> tab[i][j].y;
        }
    }
    vector<vector<int>> dist (n,vector<int> (m,1E9));
    dist[t.x][t.y]=0;
    set<pair<int,pair<int,int>>> q;
    q.insert({0,{t.x,t.y}});
```

```

while (!q.empty())
{
    cord lol=cord (q.begin()->second.first ,q.begin()->second.second);
    int bonuskekw=0;
    //cout << lol.x << " " << lol.y << endl;
    q.erase(q.begin());
    if (lol.x+tab[lol.x][lol.y].x>=0 && lol.x+tab[lol.x][lol.y].x<n)
    {
        if (lol.y+tab[lol.x][lol.y].y>=0 && lol.y+tab[lol.x][lol.y].y<m)
        {
            if (dist[lol.x+tab[lol.x][lol.y].x][lol.y+tab[lol.x][lol.y].y]>dist[lol.x][lol.y])
            {
                if (q.find({dist[lol.x+tab[lol.x][lol.y].x][lol.y+tab[lol.x][lol.y].y],{lol.x,lol.y}})!=q.end())
                {
                    q.erase(q.find({dist[lol.x+tab[lol.x][lol.y].x][lol.y+tab[lol.x][lol.y].y],{lol.x,lol.y}}));
                }
                dist[lol.x+tab[lol.x][lol.y].x][lol.y+tab[lol.x][lol.y].y]=dist[lol.x][lol.y];
                q.insert({dist[lol.x+tab[lol.x][lol.y].x][lol.y+tab[lol.x][lol.y].y],{lol.x+tab[lol.x][lol.y].x,lol.y+tab[lol.x][lol.y].y}});
            }
        }
    }
    cord tyta=cord(tab[lol.x][lol.y].x+lol.x,tab[lol.x][lol.y].y+lol.y);
    for (int i=0; i<n; i++)
    {
        for (int j=0; j<m; j++)
        {
            if (dist[i][j]>dist[lol.x][lol.y]+abs(tyta.x-i)+abs(j-tyta.y)+bonuskekw)
            {
                if (dist[i][j]!=1E9)
                {
                    q.erase(q.find({dist[i][j],{i,j}}));
                }
                dist[i][j]=dist[lol.x][lol.y]+abs(tyta.x-i)+abs(j-tyta.y)+bonuskekw;
                q.insert({dist[i][j],{i,j}});
            }
        }
    }
}
cout << dist[h.x][h.y];

return 0;
}

```

## Task E ()

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

#define return0 ios_base::sync_with_stdio(false);cin.tie(0);cout.tie(0);
#define se second
#define fi first

typedef long long ll;
typedef long double ld;
typedef unsigned long long ull;

using namespace std;

struct cord
{
    ll x;
    ll y;
    cord (ll a=-1, ll b=-1)
    {
        x=a;
        y=b;
    }
    void out()
    {
        cout << x << " " << y << endl;
    }
};

cord sumvect (cord a, cord b)
{
    return cord (a.x+b.x,a.y+b.y);
}
cord unsumvect (cord a, cord b)
{
    return cord (a.x-b.x,a.y-b.y);
}

cord vect (cord a, cord b)
{
    return cord (b.x-a.x,b.y-a.y);
}
bool operator==(cord &a, cord &b){
return (a.x==b.x && a.y==b.y);
}

bool operator<(cord &a, cord &b)
{
    if (a.x==b.x)
    {
        return a.y<b.y;
    }
    return a.x<b.x;
}

void rip (cord a, cord b){
cout << "? " << a.x << " " << a.y << " " << b.x << " " << b.y << endl;
}

int main()
{
    return0
    int n,m,b;
    cin >> n >> m >> b;

    vector <cord> tab (b);
    for (int i=0; i<b; i++){
        cin >> tab[i].x >> tab[i].y;
    }
    sort(tab.begin(),tab.end());
    cord a=tab[0],c=cord(tab[0].x+n,tab[0].y+m);
    rip (a,c);
    cord tmp;
    cin >> tmp.x >> tmp.y;
    cord ofans;
    if (tmp==a){
```

```

    ofans=cord(n+1,m+1);
    if (b==1){
        cout <<"!_" << n+1 << "_" << m+1 << endl;
    } else if (b==2) {
        cord vec=(tab[0],tab[1]);
        cord a3=sumvect(c,vec),a4=unsumvect(c,vec);
        rip(a3,a4);
        cin >> tmp.x >> tmp.y;
        if (tmp==a3){
            ofans=(unsumvect(a3,tab[0]));
        } else {
            ofans=(unsumvect(a4,tab[1]));
        }
        cout << "!_" << ofans.x << "_" << ofans.y << endl;
        //cout <<
    }
} else {
    ofans=cord(1,1);
    if (b==1){
        cout << "!_1_1" << endl;
    } else if (b==2){
        cord vec=(tab[0],tab[1]);
        cord a3=sumvect(a,vec),a4=unsumvect(a,vec);
        rip(a3,a4);
        cin >> tmp.x >> tmp.y;
        if (tmp==a3){
            ofans=(unsumvect(a3,tab[0]));
        } else {
            ofans=(unsumvect(a4,tab[1]));
        }
        cout << "!_" << ofans.x << "_" << ofans.y << endl;
        //cout <<
    } else {
        sort(tab.begin(),tab.end());
        //      ofans ();
    }
}
return 0;
}

```



## Task F ()