

## Разбор задач олимпиады «Школьная олимпиада по Красноярскому краю, 9-11 классы» сайта [aspr.ru](http://aspr.ru) от 16.10.2013 года

### Задача А. От перестановки что-то меняется...

Для решения этой задачи надо было проверить на истинность три равенства – равно ли число сумме двух других.

#### Решение на языке C++

```
#include <iostream>
#include <cstdio>

using namespace std;

int main()
{
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
    int a, b, c;
    cin>>a>>b>>c;
    if(a==b+c || b==a+c || c==a+b) cout<<"YES";
    else cout<<"NO";
    return 0;
}
```

#### Решение на языке Паскаль

```
var a, b, c : Longint;
begin
    assign(input, 'input.txt'); reset(input);
    assign(output, 'output.txt'); rewrite(output);
    Read(a, b, c);
    if (a=b+c) or (b=a+c) or (c=a+b) then Write('YES')
    else Write('NO');
end.
```

### Задача В. Развлечения гномов

Из условия задачи понятно, что надо три числа перевести из десятичной системы счисления. Чтобы не писать один код несколько раз, напомним процедуру и вызовем её трижды.

Для перевода числа из десятичной системы счисления воспользуемся алгоритмом, который изучается на уроках информатики. Будем делить число на основание системы счисления, пока не получим нуль. Ответом будут остатки от деления, записанные в обратном порядке. Если основание новой системы счисления больше 10, то возможно появление цифр больше 9, их надо обработать с помощью оператора *if*.

#### Решение на языке C++

```
#include <iostream>
#include <cstdio>
#include <stack>
```

```
using namespace std;

void fun(int a, int b){
    stack<char> s;
    while(a){
        int t = a % b;
        if(t<10) s.push((char)('0'+t));
        else s.push((char)('A'+t-10));
        a /= b;
    }
    while(!s.empty()){
        cout<<s.top();
        s.pop();
    }
}

int main(){
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
    int d, m, y;
    char c1, c2;
    cin>>d>>c1>>m>>c2>>y;
    fun(d,d+1); cout<<c1;
    fun(m,d+1); cout<<c2;
    fun(y,d+1);
    return 0;
}
```

### **Решение на языке Паскаль**

```
var
    d, m, y, code : Longint;
    str, s : string;

procedure fun(a, b : Longint);
var
    s : String;
    t : Longint;
begin
    s := '';
    while a>0 do begin
        t := a mod b;
        a := a div b;
        if t<10 then s := chr(48+t)+s
        else s := chr(ord('A')+t-10)+s;
    end;
    Write(s);
end;

begin
    assign(input, 'input.txt'); reset(input);
    assign(output, 'output.txt'); rewrite(output);
    read(str);
    val(str, d, code);
    s:=copy(str, 1, code-1);
    str:=copy(str, code+1, length(str)-code);
    val(s, d, code);
    val(str, m, code);
    s:=copy(str, 1, code-1);
    str:=copy(str, code+1, length(str)-code);
    val(s, m, code);
    val(str, y, code);

    fun(d, d+1); Write('/');
```

```
    fun(m, d+1); Write('/');
    fun(y, d+1);
end.
```

## Задача С. Звездные прямоугольники

Переберём всевозможные тройки звёзд. Если очередная тройка образует прямой угол (это легко проверить с помощью скалярного произведения векторов, натянутых на эти три точки), то нетрудно достроить их до прямоугольника. То есть однозначно найти четвёртую точку. Осталось проверить её наличие среди всего множества точек.

Циклы организуем таким образом, чтобы каждое подмножество выбиралось ровно один раз. Это позволит избежать повторений (и, как следствие, ненужных делений ответа на 6 или 24) и ускорит программу. Асимптотика времени работы программы составит  $O(N^4)$ , но это будут простые арифметические операции, а, благодаря такой организации циклов, реальное количество операций будет на порядок меньше.

### Решение на языке C++

```
#include <iostream>
#include <cstdio>

using namespace std;

int x[109], y[109];

bool check(int i, int j, int k, int &xx, int &yy){
    if((x[j]-x[i])*(x[k]-x[i])+(y[j]-y[i])*(y[k]-y[i]) == 0){
        xx = x[j] - x[i] + x[k];
        yy = y[j] - y[i] + y[k];
        return true;
    }
    if((x[i]-x[j])*(x[k]-x[j])+(y[i]-y[j])*(y[k]-y[j]) == 0){
        xx = x[i] - x[j] + x[k];
        yy = y[i] - y[j] + y[k];
        return true;
    }
    if((x[j]-x[k])*(x[i]-x[k])+(y[j]-y[k])*(y[i]-y[k]) == 0){
        xx = x[j] - x[k] + x[i];
        yy = y[j] - y[k] + y[i];
        return true;
    }
    return false;
}

int main(){
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
    int n, xx, yy, res=0;
    scanf("%d", &n);
    for(int i=0; i<n; ++i)
        scanf("%d%d", &x[i], &y[i]);

    for(int i=0; i<n; ++i)
        for(int j=i+1; j<n; ++j)
            for(int k=j+1; k<n; ++k)
                if(check(i, j, k, xx, yy))
                    for(int ii=k+1; ii<n; ++ii)
                        if(x[ii]==xx && y[ii]==yy) res++;

    cout<<res;
```

```
    return 0;  
}
```

### Решение на языке Паскаль

```
var  
    n, i, j, k, ii, res, xx, yy : Longint;  
    x, y : array [1..109] of Longint;  
  
function check(i, j, k : Longint) : Boolean;  
begin  
    check := false;  
    if (x[j]-x[i])*(x[k]-x[i])+(y[j]-y[i])*(y[k]-y[i]) = 0 then begin  
        xx := x[j] - x[i] + x[k];  
        yy := y[j] - y[i] + y[k];  
        check := true;  
    end;  
    if (x[i]-x[j])*(x[k]-x[j])+(y[i]-y[j])*(y[k]-y[j]) = 0 then begin  
        xx := x[i] - x[j] + x[k];  
        yy := y[i] - y[j] + y[k];  
        check := true;  
    end;  
    if (x[j]-x[k])*(x[i]-x[k])+(y[j]-y[k])*(y[i]-y[k]) = 0 then begin  
        xx := x[j] - x[k] + x[i];  
        yy := y[j] - y[k] + y[i];  
        check := true;  
    end;  
end;  
  
begin  
    assign(input, 'input.txt'); reset(input);  
    assign(output, 'output.txt'); rewrite(output);  
    Read(n);  
    for i:=1 to n do  
        Read(x[i], y[i]);  
  
    for i:=1 to n do  
        for j:=i+1 to n do  
            for k:=j+1 to n do  
                if check(i, j, k) then  
                    for ii:=k+1 to n do  
                        if (x[ii]=xx) and (y[ii]=yy) then inc(res);  
  
    Write(res);  
end.
```

### Задача D. Мафия в городе

Данную задачу следует решать полным перебором подмножеств базовых станций и проверкой все ли каналы связи контролируются при захвате данного подмножества станций. А среди всех подходящих надо выбрать то подмножество, в котором меньше всего вышек. Перебор подмножеств удобней всего реализовывать в двоичной системе счисления. Действительно, если перебрать все числа от 0 до  $2^N - 1$ , то мы получим все возможные варианты расстановки  $N$  нулей и единиц (где единицы будут соответствовать выбранным элементам множества):

```
000...00  
000...01  
000...10  
⋮
```

111...10

111...11

Информацию о каналах связи тоже будем хранить в двоичном представлении – отметим единичками те биты, которые соответствуют задействованным базовым станциям. Благодаря такому представлению, битовая операция «И» между выбранным подмножеством станций и каналом связи будет давать ответ на вопрос «Контролируется ли данный канал связи мафией, если захвачено это подмножество станций?».

Чтобы посчитать количество станций в выбранном подмножестве воспользуемся функцией *count*, которая подсчитывает количество единиц в двоичном представлении числа. Для этого используется конструкция  $a \& (a-1)$ , которая была нами описана в одном из предыдущих разборов.

### Решение на языке C++

```
#include <iostream>
#include <cstdio>
#include <memory.h>

using namespace std;

int n, m, mas[500], a, b;

int count(int a){
    int r=0;
    while(a){
        a&=(a-1);
        ++r;
    }
    return r;
}

int main()
{
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
    cin>>n>>m;

    for(int i=0;i<m;++i) {
        cin>>a>>b;
        mas[i] = (1<<(a-1)) | (1<<(b-1));
    }

    int res=n, j;
    a = -1;
    b = 0;

    for(int i=0;i<(1<<n);++i){
        for(j=0;j<m;++j) if((i&mas[j])==0) break;
        if(j==m){
            int t = count(i);
            if(t<res){
                res = t;
                a = i;
                b = 0;
            }
            if(t==res) ++b;
        }
    }

    cout<<res<<" "<<b<<endl;
```

```
if(res>0)
    for(int i=0;i<n;++i)
        if(a&(1<<i))
            cout<<i+1<<" ";

    return 0;
}
```

### **Решение на языке Паскаль**

```
var
    n, m, a, b, i, j, res, t : Longint;
    mas : array [1..500] of Longint;
function count(a : Longint) : Longint;
var
    r : Longint;
begin
    r := 0;
    while a>0 do begin
        a := a and (a-1);
        inc(r);
    end;
    count := r;
end;

begin
    assign(input, 'input.txt'); reset(input);
    assign(output, 'output.txt'); rewrite(output);
    Read(n, m);

    for i:=1 to m do begin
        Read(a, b);
        mas[i] := (1 shl (a-1)) or (1 shl (b-1));
    end;

    res := n;
    a := -1;
    b := 0;

    for i:=0 to (1 shl n)-1 do begin
        j := 1;
        while (j<=m) and ((i and mas[j])>0) do
            inc(j);
        if j>m then begin
            t := count(i);
            if t < res then begin
                res := t;
                a := i;
                b := 0;
            end;
            if t=res then inc(b);
        end;
    end;

    WriteLn(res, ' ', b);
    if res>0 then
        for i:=0 to n-1 do
            if (a and (1 shl i))<>0 then
                Write(i+1, ' ');
end.
```