

Meno: Michal Korbela
trieda: sexta
škola: Gymnázium J.J. Bánovce
úloha č.5

Po dlhom a namáhavom rozmýšľaní som na to konečne došiel. Každý vie, že každý banán je zahnutý - je nekonvexný, a každý pomaranč je konvexný. Takto však vyzerajú len ideálne banány a pomaranče a zďiaľky. Keď sa však pozrieme zblýska, tak môžu mať stopku, ktorá spraví z konvexného pomaranča nekonvexný útvar. Takže týmto smerom cesta nevedie. Čo však ešte odlišuje banán od pomaranča?? pomaranč sa dá teoreticky vpísať do štvorca - trošku škaredší do obdĺžnika s pomerom strán blízko 1. Ak by sa banán dal vpísať do takéhoto útvaru, tak by veľa políčok zostalo bielych - to je kľúčová myšlienka. Ešte je tu jeden problém, ako spočítať čierne políčka. Niektoré útvary však môžu byť duté, tak to musíme nejako obísť. Útvar si zafarbíme. A to tak, že ak bude nejaké políčko biele, tak otestujeme, či zo všetkých strán je čierne políčko, keďže povrch je súvislý. Testovať budeme tak, že sa pustíme najskôr doľava od neho, potom doprava, a nakoniec hore a dole. Ak aspoň v jedno teste dôjdeme na okraj obrázka, tak vieme, že políčko nepatrí útvaru. Musíme vyskúšať všetky smery, z dôvodu, že sú tu banány, ktoré môžu byť otočené hociako. Potom Všetky pomaranče bud mať pomer čiernych väčší ako 1:1 - keď si vypočítame obsah kruhu, tak to výjde okolo 3/4. Ešte vždy budeme testovať obsah útvaru vo štvorci, aby sme zväčšili pomer čiernych políčok ku bielym. Aby sme netestovali všetky biele políčka, tak si všetky biele rady vymažeme. Aby som nezabudol, tak časová zložitosť je max $O(r.s.(r+s))$, lebo to je časová zložitosť toho testovania, ale keďže je v obrázku nejaký útvar, tak to bude rádovo menej. A pamäťová - $O(r.s)$ - zapamätáme si len čierne a biele body.

```
#include <iostream>
```

```
#include <stdio.h>
```

```
using namespace std;
```

```
bool pole[2100][2100];  
long x1;  
long x2;  
  
long y2;  
long y_1;
```

```
long test(long y, long x){                                     //funkcia testovanie -  
    otestuje, či je daný bod súčasťou útvaru, alebo nie  
    long f=1;  
    long g=0;  
    for(long i=x+1; i<=x2; i++)                               //ideme doprava  
        if(pole[y][i]==1){ g=1; break; }  
    f*=g;  
  
    g=0;  
    if(f==1){  
        for(long i=x-1; i>=x1; i--)                             //ideme doľava  
            if(pole[y][i]==1){ g=1; break; }  
    }
```

```

        f*=g;
    }
g=0;
    if(f==1){
        for(long i=y-1; i>=y_1; i--)
            if(pole[i][x]==1){ g=1;break;}
        f*=g;
    }
    if(f==1){
g=0;
        for(long i=y+1; i<=y2; i++)
            if(pole[i][x]==1){ g=1;break;}
        f*=g;
    }
    return f;}
súčasťou tvaru, 0 - nie je súčasťou
//ideme hore
//ideme dole
//vráti hodnotu 1 - bod je

```

```

//////////////////////////////////////////////////main//////////////////////////////////////
///
int main(){

    long lama;
    cin>>lama;
    for(long p=1; p<=lama;p++){

x1=0;x2=0;y_1=0;y2=0;
long a=0;
long b=0;

        for(long i=0; i<=2050;i++){
            for(long il=0; il<=2050;il++){
                pole[i][il]=0;}
            }
        long s,r;

        cin>>s>>r;

        for(long i=1; i<=r;i++)
            for(long il=1; il<=s;il++)
                scanf("%d",&pole[i][il]);
        premenných
//nacitanie
        while(0==0){
            b=1;
            while(pole[a][b]==0 && b<=s)
                b++;

```

```

        if( pole[a][b]!=0 ){y_1=a; break;}
    a++;
}
//nájdienie y súradnicu horného
ľavého bodu

a=r;
b=s;
while(0==0){
    b=s;
    while(pole[a][b]==0 && b>0)
        b--;
    if( pole[a][b]!=0 ){y2=a; break;} // nájdienie y
    súradnice dolného ľavého bodu
    a--;
}

a=r;
b=s;
while(0==0){
    a=r;
    while(pole[a][b]==0 && a>0) //nájdienie x
        a--;
    súradnicu dolného pravého bodu
    if( pole[a][b]!=0 ){x2=b; break;}
    b--;
}

a=1;
b=1;
while(0==0){
    a=1;
    while(pole[a][b]==0 && a<=r) //nájdienie x
        a++;
    súradnicu horného ľavého bodu
    if( pole[a][b]!=0 ){x1=b; break;}
    b++;
}

//find
int err=0;
long long sum=0;

for(long i=x1; i<=x2;i++)
for(long il=y_1; il<=y2; il++){
    bodov v obdĺžniku, kde sa útvar nachádza //testovanie len
    if(pole[il][i]==0 && test(il,i)==1) sum++; // ak je bod súčasťou
    tvaru
}

```

```

else if(pole[i1][i]==1) sum++;           // ak je bod 1, tak
je určite súčasťou útvaru
    }

long q=0;
if(x2-x1>y2-y_1)q=sum*100/((x2-x1+1)*(x2-x1+1)); //vypočítame pomer
čiernych ku bielym
else q=sum*100/((y2-y_1+1)*(y2-y_1+1));

if(q<50) err=1;           // aj je počet čiernych bodov viac ako 50%,
tak je to pomaranč ak nie, tak je to banán

if(err==1) cout<<"banan"<<endl;
else cout<<"pomaranc"<<endl;
}

}

```