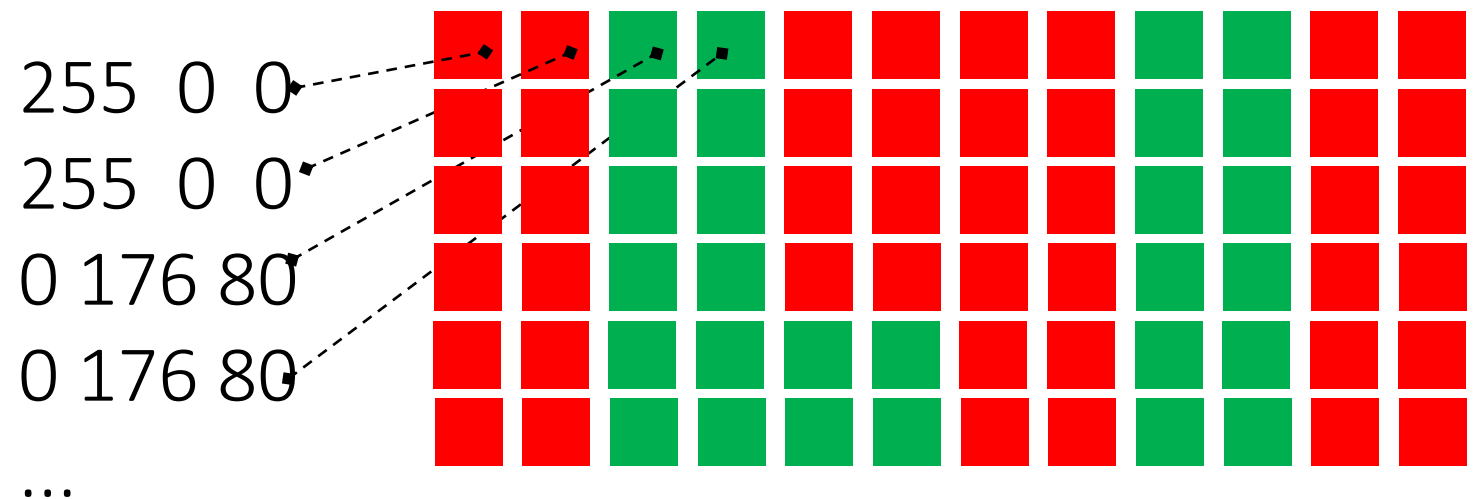


[illegible]

Egy digitális kép tárolásánál minden egyes képpont színét tároljuk. A képpontok színét az RGB kód adja. Ezen színösszetevők értéke 0 és 255 közötti egész szám lehet.

A *kep.txt* fájlban egy 50×50 képpontos kép képpontjainak RGB kódjai vannak

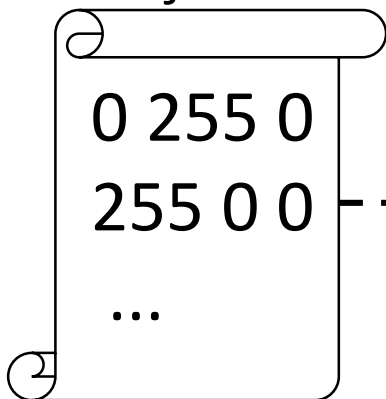
Az állomány a képet sorfolytonosan, a képpontok RGB kódját szóközzel elválasztva tartalmazza, minden képpontot egy újabb sorban



1. Olvassa be a fájlból egy megfelelő adatszerkezetbe az képpontok RGB kódját!  
másolás
2. Egy tetszőleges (input) RGB kódú pixel megtalálható-e a képen, ha igen hol?  
lineáris keresés
3. A kép 35. sor 11. képpontjának színe milyen RGB kódú és hányszor szerepel a 35. sorban, illetve a 11. oszlopban.  
kiválasztás, megszámlálás
4. A vörös, zöld és kék színek közül melyik szín fordul elő legtöbbször a képen?  
megszámlálás (tömbbe), maximum kiválasztás



**f1():** Olvassa be a fájlból egy megfelelő adatszerkezetbe az képpontok RGB kódját



```
0 255 0
255 0 0
...
```

<Pixel> *pixelek* :lista  
másolás-tétele

**Pixelek** osztály

- rgb :String
- sor :int
- oszlop :int

A főprogramban **ciklusokkal** a fájl beolvasása és a **Pixel** osztály konstruktorának hívása, aki feltölti az rgb adatmezőt,  
mivel a kép 50 x 50 pixeles és a pontok színe és sorfolytonos ezért 2 ciklus kell  
külső ciklus a sor 50-szer  
belső az oszlop 50-szer  
be: *px* → *konstruktor(px, sor, oszlop)*



f2()

Egy tetszőleges (input) RGB kódú pixel megtalálható-e a képen, ha igen hol? `px(x, y)`

A főprogramban a *pixelek* elemszáma ismert, a **Pixel** osztályt a `getRgb()` metódussal kell lekérdezni

Ha találat, akkor

Ki: `getSor()`, `getOszlop()`

<Pixel> *pixelek* :lista  
lineáris keresés

**Pixelek** osztály

- rgb :String
- sor :int
- oszlop :int
- +getRgb()
- +getSor()
- +getOszlop()



**f3()**

A kép px(35, 11) képpontjának színe milyen RGB kódú és hányszor szerepel a 35. sorban, illetve a 11. oszlopban.

A végig a *pixeleken*, amíg a  
getSor() és a getOszlop() nem lesz jó ☺  
akkor: **px**=getRgb()

újra végig a *pixeleken*

ha a getSor=35 és **px**=getRgb() akkor sDb++  
ha a getOszlop=11 és **px**=getRgb() akkor oDb++

<Pixel> *pixelek* :lista  
kiválasztás, megszámlálás

**Pixelek** osztály

- rgb :String
- sor :int
- oszlop :int
- +getRgb()
- +getSor()
- +getOszlop()



**f4()**

A vörös, zöld és kék színek közül melyik szín fordul elő legtöbbször a képen?

```
rgbDB[] = {0, 0, 0};
```

A végig a *pixeleken*,

elágazás: getRgb() szerint

ha "255 0 0" :rgbDB[0]++

ha "0 255 0" :rgbDB[1]++

ha „0 0 255" :rgbDB[2]++

```
color[] = {"vörös", "zöld", "kék"};
```

maximum kiválasztás az rgbDB[]-ben → color[]

<Pixel> *pixelek* :lista  
kiválasztás, megszámlálás

**Pixelek** osztály

-rgb :String

-sor :int

-oszlop :int

+getRgb()

+getSor()

+getOszlop()



Az if-else ág helyett egy olyan vizsgálat, ahol egy változó értékeit közvetlenül lehet vizsgálni, és azokhoz más-más utasításokat rendelhetnek hozzá.

```
switch (szín) {  
    case "255 0 0":  
        rgbDB[0]++;  
        break;  
    case "0 255 0":  
        rgbDB[1]++;  
        break;  
    case "0 0 255":  
        rgbDB[2]++;  
        break;  
}
```

