



International Olympiad in Informatics 2013

6-13 July 2013

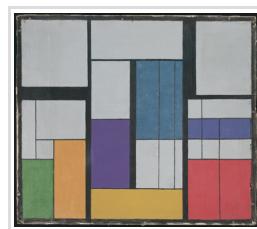
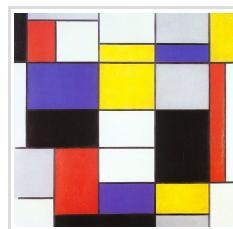
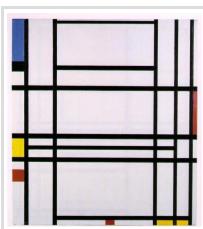
Brisbane, Australia

**art
class**
Azeri — 1.1

İncəsənət tarixindən imtahan yaxınlaşır, ancaq Siz məktəbdə incəsənətə deyil, informatikaya daha çox diqqət yetirirsiz! Siz elə program yazmalısınız ki, Sizin əvəzinizə imtahan versin.

İmtahanda bir neçə rəsm əsəri olacaq. Hər bir rəsm 1, 2, 3 və 4 kimi nömrələnmiş dörd fərqli üslubdan birinə aiddir.

Üslub 1 neoplastik modern incəsənətə aiddir. Məsələn:



Üslub 2 impressionist mənzərəyə incəsənətə aiddir. Məsələn:



Üslub 3 ekspressionist hərəkət rəsmlərinə aiddir. Məsələn:



Üslub 4 rəngli sahə rəsmlərinə aiddir. Məsələn:



Siz verilmiş rəqəmsal rəsm görüntüsünə görə, onun hansı üsluba aid olmasını müəyyənləşdirməkdir.

IOI hakimləri hər bir üsluba aid çoxlu görüntü toplamışlar. Təsadüfi qaydada hər bir üslubdan 9 görüntü seçilib və məsələ materialı kimi Sizin kompüterə daxil edilib. Belə ki, Siz onları əllə yoxlaya və testləmə üçün istifadə edə bilərsiniz. Qalan görüntülər kompüterinə qiymətləndirmə zamanı ötürülləcək.

Hər bir görüntü piksellərin $H \times W$ toru şəklində veriləcək. Görüntünün sətirləri yuxarıdan aşağıya $0, \dots, (H - 1)$ kimi, sütunları isə soldan sağa $0, \dots, W - 1$ kimi nömrələnib.

Piksəllər ikiölçülü R , G və B massivləri vasitəsilə təsvir olunub. Bu massivlər görüntünün hər bir pikselindəki uyğun olaraq qırmızı, yaşıl və göy rəngin miqdarını göstərir. Bu kəmiyyətlər 0 -dan (qırmızı, yaşıl və ya göy yoxdur) 255 -dək (qırmızı, yaşıl və ya göy maksimal miqdardadır) aralıqdadır.

Implementation

You should submit a file that implements the function `style()`, as follows:

Your Function: `style()`

C/C++

```
int style(int H, int W,
          int R[500][500], int G[500][500], int B[500][500]);
```

Pascal

```
type artArrayType = array[0..499, 0..499] of longint;
function style(H, W : LongInt;
               var R, G, B : artArrayType) : LongInt;
```

Description

This function should determine the style of the image.

Parameters

- H : The number of rows of pixels in the image.
- W : The number of columns of pixels in the image.
- R : A two-dimensional array of size $H \times W$, giving the amount of red in each pixel of the image.
- G : A two-dimensional array of size $H \times W$, giving the amount of green in each pixel of the image.
- B : A two-dimensional array of size $H \times W$, giving the amount of blue in each pixel of the image.
- *Returns*: The style of the image, which must be 1 , 2 , 3 or 4 , as described above.

Each array element $R[i][j]$, $G[i][j]$ and $B[i][j]$ refers to the pixel in row i and column j , and will be an integer between 0 and 255 inclusive.

Constraints

- Time limit: 5 seconds
 - Memory limit: 64 MiB
 - $100 \leq H \leq 500$
 - $100 \leq W \leq 500$
-

Scoring

There are no subtasks. Instead, your score for this task will be based on how many images your program correctly classifies.

Suppose you correctly classify P percent of the images (so $0 \leq P \leq 100$):

- If $P < 25$ then you will score 0 points.
 - If $25 \leq P < 50$ then you will score between 0 and 10 points, on a linear scale. Specifically, your score will be $10 \times (P - 25) / 25$, rounded down to the nearest integer.
 - If $50 \leq P < 90$ then you will score between 10 and 100 points, on a linear scale. Specifically, your score will be $10 + (90 \times (P - 50) / 40)$, rounded down to the nearest integer.
 - If $90 \leq P$ then you will score 100 points.
-

Experimentation

The sample grader on your computer will read input from the file `artclass.jpg`. This file must contain an image in JPEG format.

You are allowed to use any available graphics processing applications to study the images, but this is not necessary to solve the problem. (See the menu "Applications > Graphics".)

Language Notes

C/C++ You must `#include "artclass.h"`.

Pascal You must define the `unit ArtClass`. All arrays are numbered beginning at `0` (not `1`).

See the solution templates on your machine for examples.