



International Olympiad in Informatics 2013

6-13 July 2013

Brisbane, Australia

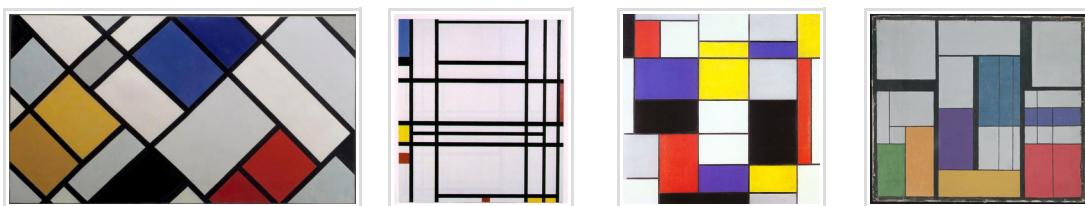
**art
class**

Deutsch — 1.1

Bald hast du eine Prüfung in Kunstgeschichte, aber du hast in der Schule der Informatik mehr Aufmerksamkeit geschenkt als der Kunstgeschichte! Du wirst ein Programm schreiben, das die Prüfung für dich ablegt.

Die Prüfung besteht aus mehreren Bildern. Jedes Bild ist ein Beispiel für eine von vier verschiedenen Stilrichtungen, nummeriert 1, 2, 3 und 4.

Stil 1 enthält Werke des Neo-Plastizismus. Zum Beispiel:



Stil 2 enthält impressionistische Landschaften. Zum Beispiel:



Stil 3 enthält expressionistische Action Paintings. Zum Beispiel:



Stil 4 enthält Werke der Farbfeldmalerei. Zum Beispiel:



Deine Aufgabe ist, von einem in digitaler Form gegebenen Bild zu bestimmen, zu welcher Stilrichtung das Bild gehört.

Die IOI Judges haben viele Bilder von jeder Stilrichtung gesammelt. Pro Stilrichtung sind zufällig neun Bilder ausgewählt worden und sind im Material auf deinem Computer enthalten, damit du sie von Hand untersuchen und fürs Testen verwenden kannst. Die restlichen Bilder wird dein Programm während des Gradings erhalten.

Das Bild wird gegeben als „Gitter“ von $H \times W$ Pixeln. Die Zeilen des Bildes sind von oben nach unten mit $0, \dots, (H - 1)$ nummeriert, und die Spalten sind von links nach rechts mit $0, \dots, W - 1$ nummeriert.

Die Pixel werden mit den zweidimensionalen Arrays `R`, `G` und `B` beschrieben, welche den Rot-, Grün- und Blauanteil jedes Pixels des Bildes angeben. Diese Werte gehen von 0 (kein Rot, Grün oder Blau) bis 255 (der maximale Wert).

Implementierung

Du musst die Funktion `style()` wie folgt implementieren:

Deine Funktion: `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;
```

Beschreibung

Die Funktion soll die Stilrichtung des Bildes bestimmen.

Parameter

- H : Die Anzahl Zeilen des Bildes.
- W : Die Anzahl Spalten des Bildes.
- R : Ein zweidimensionales Array der Grösse $H \times W$ mit dem Rotanteil des jeweiligen Pixel des Bildes.
- G : Ein zweidimensionales Array der Grösse $H \times W$ mit dem Grünanteil des jeweiligen Pixel des Bildes.
- B : Ein zweidimensionales Array der Grösse $H \times W$ mit dem Blauanteil des jeweiligen Pixel des Bildes.
- *Rückgabe*: Die Stilrichtung des Bildes ($1, 2, 3$ oder 4) wie oben beschrieben.

Jedes Arrayelement $R[i][j]$, $G[i][j]$ und $B[i][j]$ bezieht sich auf das Pixel in Zeile i und Spalte j und ist jeweils eine ganze Zahl zwischen 0 und 255 inklusive.

Beschränkungen

- Zeitlimit: 5 Sekunden
 - Speicherlimit: 64 MiB
 - $100 \leq H \leq 500$
 - $100 \leq W \leq 500$
-

Bewertung

Die Aufgabe hat keine Subtasks. Deine Punktzahl hängt ausschliesslich davon ab, wieviele Bilder dein Programm korrekt einteilt.

Sei P der Anteil der Bilder (in Prozent), die dein Programm korrekt einteilt (also $0 \leq P \leq 100$):

- Falls $P < 25$ dann erhältst du 0 Punkte.
 - Falls $25 \leq P < 50$ dann erhältst du zwischen 0 und 10 Punkte, linear interpoliert. Deine Punktzahl ist demnach $10 \times (P - 25) / 25$, abgerundet auf die nächst kleinere ganze Zahl.
 - Falls $50 \leq P < 90$ dann erhältst du zwischen 10 und 100 Punkte, linear interpoliert. Deine Punktzahl ist demnach $10 + (90 \times (P - 50) / 40)$, abgerundet auf die nächst kleinere ganze Zahl.
 - Falls $90 \leq P$ dann erhältst du 100 Punkte.
-

Testen

Der Beispiel-Grader auf deinem Computer liest von der Eingabedatei `artclass.jpg`. Die Datei muss ein Bild im JPEG Format sein.

Du darfst ein beliebiges Bildbearbeitungsprogramm verwenden, um die Bilder zu studieren, aber das ist nicht nötig, um das Problem zu lösen. (Siehe Menü "Applications > Graphics".)

Sprachspezifische Bemerkungen

C/C++ Du musst `#include "artclass.h"` verwenden.

Pascal Du musst `unit ArtClass` definieren. Alle Arrays sind mit 0 beginnend indiziert. (nicht 1).

Siehe Lösungsvorlagen auf deiner Maschine für Beispiele.