



CFGS DESENVOLUPAMENT D'APLICACIONS MULTIPLATAFORMA – LOGÍSTICA			
MÒDUL 6: Accés a dades			
NOM I COGNOMS:			Nota:
UF1: Persistència en fitxers	CURS: 21 - 22	DAM-L segon curs	

Resultats d'aprenentatge:

- Dissenya, prova i documenta programes que realitzen diferents operacions sobre fitxers, documentant el programa i les proves realitzades.

Criteris de puntuació de la pràctica

- La pràctica perdrà un punt si l'informe no està entregat en pdf (si cal ser entregat).
- La pràctica pot perdre fins a un punt si no hi ha comentaris al codi.

Desenvolupament de la pràctica

- Les instruccions per realitzar la pràctica estan detallades a cada apartat.
- Haurà de ser entregat aquest informe (en pdf) i els fitxers de classe (.java) generats a la pràctica (en els casos que correspongui). (**codi → projecte Netbeans(ZIP) , imatges al PDF**)
- Haureu de fer les proves adients a cada un dels apartats especificant els paràmetres d'entrada i els resultats en aquest informe.

PRÀCTICA 2 – LECTURA/ESCRITURA DE FITXERS D'IMATGES

1. Descarrega una imatge **JPG** d'internet, cal que la imatge sigui en color. També pots emprar qualsevol imatge teva. Millor si la imatge no és molt pesada. (**1 punt**)
2. Utilitza el que has après a classe per generar tres imatges separades partint de la original. Les imatges s'han de correspondre als canals **R**, **G** i **B** de la imatge original **ARGB**. (**2 punts**)
3. Els valors RGB es converteixen a escala de grisos mitjançant, per exemple, la fórmula **NTSC**: **0,299 · Vermell (R) + 0,587 · Verd (G) + 0,114 · Blau (B)**. Aquesta fórmula representa de prop la percepció relativa de la brillantor de la llum vermella, verda i blava de la persona mitjana. Aplicant la fórmula anterior, genera una imatge **ARGB** (realment en color) que es veu com si fos en escala de grisos. El canal **Alpha** el pots deixar igual. (**3 punts**)

NOTA: Després de fer la conversió només ha de crear un nou color amb les tres components igual a la suma que has calculat, i a la nova imatge introduir el nou color per a cada pixel (i,j):

```
int sum = red+green+blue;
Color newColor = new Color(sum,sum,sum);
image.setRGB(i,j,newColor.getRGB());
```

	PC4_INF_PRAC_2	4/10/21	Pràctica 2ª DAM M06. Accés a dades		
			Professors	ramonsoler.prof	Pàgina 1 de 2



CFGS DESENVOLUPAMENT D'APLICACIONS MULTIPLATAFORMA – LOGÍSTICA			
MÒDUL 6: Accés a dades			
NOM I COGNOMS:			Nota:
UF1: Persistència en fitxers	CURS: 21 - 22	DAM-L segon curs	

NOTA 2: Realment ara només caldria guardar 1 byte per pixel per les tres components **RGB**, doncs esdevenen les mateixes per la imatge en escala de grisos. Això es pot aconseguir fent un **BufferedImage** nou de tipus **TYPE_BYTE_GRAY** en lloc de **TYPE_INT_ARGB**. Amb aquest nou tipus cada pixel ocupa 16 bits, 8 pel canal Alpha i 8 per la escala de gris. No obstant, no cal que ho feu, per aquesta pràctica podeu fer servir el **ARGB**.

4. Ara farem un filtre 'blur', això es un efecte de **desenfocament**. Es pot fer de manera senzilla si substitueixes cada pixel (per cada component **RGB**), per la mitjana de un entorn quadrat centrat en el pixel. L'efecte de desenfocament es pot transformar en altres similars si es programa de determinada manera, fent el que es diu una **CONVOLUCIÓ** al camp del tractament del senyal. A la imatge de abaix podeu veure un exemple, la nova posició del 31 esdevé 92, i s'obté **multiplicant cada component de l'àrea al quadrat blau per la casella corresponent al quadrat 3x3 de la dreta i sumant els resultats**. Per fer el desenfocament, totes les caselles del quadrat de la dreta han de valer **1/9**, això es **1/#caselles**.

7	23	50	64	14
15	13	31	46	8
42	25	92	31	32
71	44	74	94	92
2	43	51	35	4

 $\times$

0	2	0
0	0	0
0	0	0

 $=$

-	-	-	-	-
-	46	100	128	-
-	26	62	92	-
-	50	184	62	-
-	-	-	-	-

Observa que pots tenir un problema als extrems de la imatge original, ja que no tens totes les caselles disponibles. Per solucionar aquest problema es fa el que es diu un **padding**, es considera que a les caselles fora del limit de la imatge hi ha **0**, o el màxim valor **255**, o un **reflex** del que tens a la imatge si el límit es veu com un mirall (el **millor padding segurament**). Com a **exemple del reflex** imagina que el quadrat blau estigues centrat al 32, una posició més a la dreta, en aquest cas considerariem que a la dreta del 32 hi ha un altre 31 (anem cap a darrere). Amb tot això has de fer el següent:

- Un filtre 'blur' 5x5 (no 3x3 com a l'exemple), amb padding tipus reflex, sobre la imatge original. (2 punts)
- El mateix sobre la imatge en escala de grisos (calcula només una vegada !). (2 punts)

	PC4_INF_PRAC_2	4/10/21	Pràctica 2ª DAM M06. Accés a dades		
			Professors	ramonsoler.prof	Pàgina 2 de 2