

Interazione e Multimedia – Laboratorio Processing

Prova 3 ottobre 2016

Lo studente svolga gli esercizi richiesti usando Processing. E' suggerito l'uso del solo editor interno di Processing e delle Reference fornite con l'editor. Non sono richieste librerie esterne. Il tempo a disposizione è di **3 ore**.

*Al termine dell'esame dovrà essere consegnato un unico file zip/rar denominato con il seguente formato: **COGNOME_NOME_MATRICOLA.zip**.*

Lo zip dovrà contenere una cartella per ciascun esercizio svolto con ogni dato e file necessario alla corretta esecuzione dello sketch sviluppato.

[15] Esercizio 1

Si implementino in Processing le seguenti **classi** con le caratteristiche dichiarate :

- Una classe **Bubble** – Un Bubble è identificato da una **posizione**, da una **velocità** e da un **diametro D**. L'aspetto di un **Bubble** è di un cerchio celeste di **diametro D**. E' presente un riflesso rappresentato da un cerchio bianco di diametro pari a **un quarto di D**, posizionato in alto a sinistra nel cerchio celeste.

Il **Bubble** deve avere un metodo che gli permetta di muoversi lungo Y, alla **velocità** specificata, ad ogni ciclo di draw. Lo spostamento avverrà solo dal basso verso l'alto.

- Una classe **Fish** – Un Fish è identificato da una **posizione**, da una **velocità lungo x**, una **velocità lungo y** e un **colore**. Ha inoltre l'aspetto di un **ellisse** con asse orizzontale lungo **80 pixel** e asse verticale lungo **50 pixel**. E' presente pure una coda rappresentata da un triangolo isoscele di **altezza** pari a **40 pixel** e base di **50 pixel**. Il vertice da cui parte l'altezza del triangolo si trova all'intersezione (sinistra) tra l'asse orizzontale dell'ellisse e il suo perimetro. La base invece è perpendicolare all'asse verticale. Infine è presente un occhio nero di diametro **5 pixel** nella parte destra dell'ellisse.

Il Fish deve prevedere un metodo che gli permetta di muoversi lungo X e lungo Y alle rispettive velocità. Comunque lo spostamento lungo X sarà sempre da sinistra verso destra.

Si preveda pure un metodo che permetta di generare un **Bubble** approssimativamente sulla parte destra del Fish. Il Bubble creato avrà inoltre **diametro casuale** tra **10 e 30 pixel** e velocità pari ad un **decimo del diametro**.

Si implementi uno sketch Processing in cui:

- Si apre una finestra di dimensione 500×500, con sfondo azzurro, e **framerate 60**. Ad ogni ciclo di draw, la finestra viene aggiornata e tutti i Bubble salgono in base alla loro velocità, mentre i Fish si spostano da sinistra a destra.
- Durante ogni ciclo di draw, ogni Fish presente nella scena deve poter generare un nuovo Bubble con probabilità del 2%.
- Quando si preme il tasto "N" viene generato un nuovo Fish in posizione X uguale a -50, posizione Y casuale compresa tra 50 e 450, velocità lungo X casuale compresa tra 1 e 4, velocità lungo Y casuale compresa tra -1.5 e 1.5, colore casuale.

[15] Esercizio 2

Si implementino in Processing le seguenti **funzioni** con le caratteristiche dichiarate :

- **uniQuant()** – Deve avere come parametro di input un’immagine a scala di grigi a 8 bit (**PImage**) e un **intero k**. L’immagine di output è una nuova immagine a scala di grigi 8 bit ottenuta quantizzando l’immagine di input in maniera uniforme a **k** livelli. L’immagine quantizzata deve essere normalizzata opportunamente nell’intervallo [0, 255] prima di essere restituita come output.
- **logQuant()** – Deve avere come parametro di input un’immagine a scala di grigi a 8 bit (**PImage**) e un **intero k**. L’immagine di output è una nuova immagine a scala di grigi 8 bit ottenuta quantizzando l’immagine di input in maniera logaritmica a **k** livelli (va bene il logaritmo in base *e* di processing). L’immagine quantizzata deve essere normalizzata opportunamente nell’intervallo [0, 255] prima di essere restituita come output.

Si implementi uno sketch Processing in cui:

- Si apre un’immagine di dimensioni **256×256** (oppure si ridimensiona in modo che diventi di dimensioni **256×256**) e una finestra di dimensioni **768×280**. L’immagine deve essere a scala di grigi, se non lo è convertirla usando l’opportuna funzione di Processing. Si inizializzi inoltre una variabile **N con valore di default pari a 10**.
- Nella parte sinistra della finestra viene visualizzata l’immagine originale (a scala di grigi). Nella parte centrale viene visualizzata l’immagine ottenuta applicando **uniQuant()** all’immagine originale con parametro **N**. Nella parte destra viene visualizzato il risultato di **logQuant()** applicato all’immagine originale con parametro **N**. Sotto le tre immagini si riportino i testi “Originale”, “Uniforme”, “Logaritmica”, e il numero **N (dove si preferisce)**.
- Alla pressione del tasto ‘+’ il valore di N viene aumentato di **1**. Alla pressione del tasto ‘-’ il valore di N viene decrementato di **1**. Si limiti comunque il valore di N tra **2 e 256** (estremi compresi).