

# Interazione e Multimedia – Laboratorio Processing

## Prova 01 febbraio 2019 – Compito A

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 **80 minuti**.

*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.*

### Esercizio 1

- a) [10] Implementare una classe **Ball** con le seguenti caratteristiche: la classe è identificata da una **posizione**, da un raggio **R** e da una **velocità sy lungo y**. Di default **sy=5**. Appare come un cerchio **rosso, senza bordi** di raggio **R**. Ha un metodo per cambiare la sua posizione lungo l'asse **y** di un valore pari a **sy** per ciclo di draw. Quando il centro di **Ball** tocca il bordo superiore o inferiore della finestra la sua velocità cambia segno.
- b) [5] Estendere tramite **ereditarietà** la classe **Ball** definendo **HyperBall**. **HyperBall** differisce da **Ball** poiché possiede **anche** un attributo **colore** – di default **blu**. Deve avere un metodo che controlla se il puntatore del mouse si trova all'interno della circonferenza che lo rappresenta e, se ciò accade, il colore di **HyperBall** viene modificato in un colore casuale.

#### [OBBLIGATORIO PER VALUTARE I PUNTI PRECEDENTI]

Alla pressione di **"R"** o **"r"** lo sketch viene resettato. Implementare uno sketch in cui si apre una finestra di dimensioni **512×256**. Ad ogni ciclo di draw la finestra viene aggiornata.

**Per rendere (a) valutabile** - Nella metà sinistra istanziare un oggetto di tipo **Ball** che si muove in base al comportamento definito. La posizione iniziale è casuale e compresa nella metà sinistra della finestra, mentre il raggio è **40**.

**Per rendere (b) valutabile** - Nella parte destra della finestra va istanziato un oggetto di tipo **HyperBall** con le stesse caratteristiche al punto (a); la differenza è che il colore cambia se il puntatore del mouse vi passa sopra.

### Esercizio 2

- c) [10] Implementare la funzione con prototipo **PImage hash(PImage I, int h, int k)**. Input e output sono a scala di grigi. L'output è una nuova immagine uguale alla prima, ma in cui i pixel delle righe di indice **h** e **k**, e delle colonne di indice **h** e **k**, vengono rimpiazzati con pixel **neri**.
- d) [5] Implementare la funzione con prototipo **PImage hash(PImage I, int h, int k, int n)**. Come la precedente, ma prima di aggiungere i pixel neri, applica l'operatore massimo con finestra di dimensioni **n × n**.

#### [OBBLIGATORIO PER VALUTARE I PUNTI PRECEDENTI]

Alla pressione di **"R"** o **"r"** lo sketch viene resettato. Implementare uno sketch in cui si apre una finestra di dimensioni **768×256**. Si apre inoltre un'immagine **Im**, si converte a scala di grigi e si ridimensiona a **256×256**. Si visualizza infine questa immagine nella parte sinistra della finestra.

**Per rendere (c) valutabile** - Nella parte centrale della finestra, si visualizza l'immagine ottenuta usando **hash()** con input l'immagine originale **Im**, e con **h** e **k** casuali scelti tra 1 e 256.

**Per rendere (d) valutabile** - Nella parte destra della finestra, si visualizza l'immagine ottenuta usando **hash()** con input l'immagine originale **Im**, con **h** e **k** casuali scelti tra 1 e 256 e con **n** casuale tra 3 e 11.