



Tecnologie digitali per il suono e l'immagine 2020/21

Vincenzo Lombardo
Corso di Laurea in DAMS
Università di Torino

Mutuato in parte da Elaborazione audio e musica
(Laurea Magistrale di Informatica)

Introduzione al corso 2020-21

Tre macro argomenti

Audio Digitale

Audacity
MuseScore

Rappresentazione
Editing
Elaborazione
Archiviazione
Compressione

Immagini Digitali

GIMP
OpenShot video editor

Progettazione e
programmazione
di ambienti
multimediali
interattivi

Processing
p5*

Programma d'esame

- Suono: Cenni di acustica, Percezione uditiva, Rappresentazione digitale, Compressione del suono, Formati e supporti, Il protocollo MIDI, Esercitazioni su editor di suono e musica (Audacity, MuseScore).
- Immagine: Cenni di fisica della luce, Percezione visiva, Digitalizzazione delle immagini, Filtri, Colore, Compressione e formati, Cenni di tecnologia Video, Esercitazione su editor di immagini (GIMP, Open Shot Video Editor).
- Programmazione multimediale interattiva in Processing e P5*.

Libri di testo

- Lombardo, A. Valle, *Audio e multimedia, IV edizione*, Apogeo Education, Maggioli Editore, Milano, 2014 (Capp. 1, 2, 3, 4, 5, 6, 8).
- M. Coriasco, O. Rampado, N. Balossino, S. Rabellino, *L'immagine digitale in diagnostica per immagini*, Springer, 2013, 209 pagg., ISBN 978-88-470-5364-9, tutti i capitoli tranne il 5.

Importanza dei libri di testo



Modalità d'esame

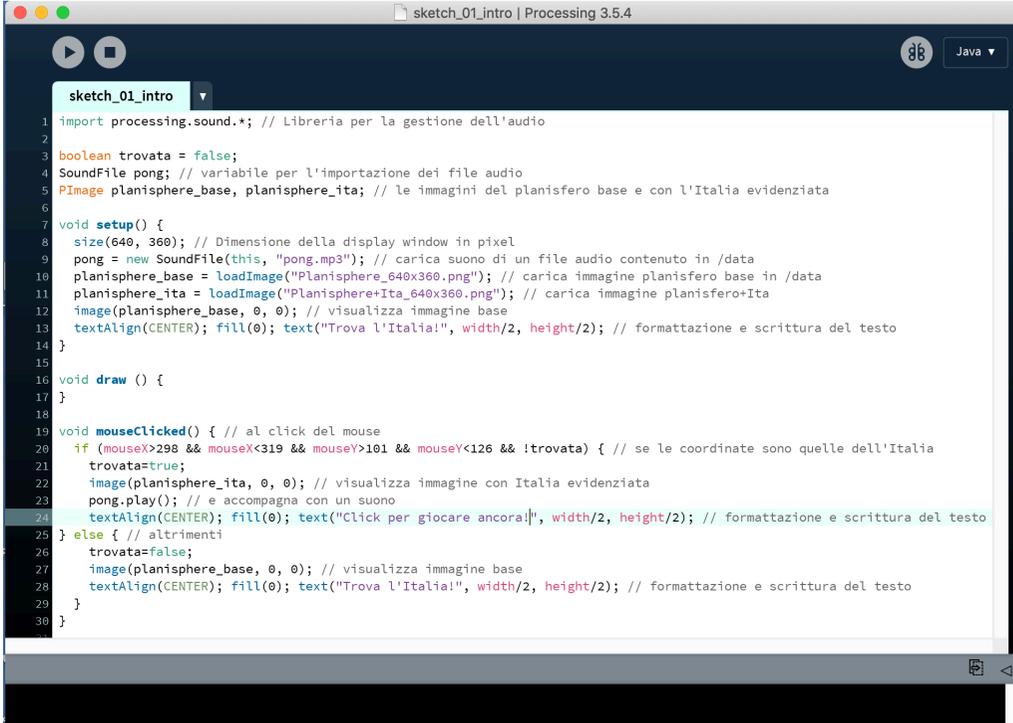
- Progetto assegnato su un tema
 - Da consegnare nelle modalità indicate una settimana prima del colloquio orale
 - Accompagnato da una relazione scritta che ne illustra gli aspetti principali (con formato indicato)
- Colloquio orale
 - Esercizio di modifica del laboratorio richiesto (quindi venire al colloquio con laptop, programmi, e file sorgenti del progetto)
 - Aspetti teorici verificati mediante esame orale

Introduzione all'ambiente Processing

Scaricare da <https://www.processing.org>

Motivazioni per Processing: visual, movimento, interazione

- Programmazione in contesto multimediale
- Linguaggio testuale (come i comuni linguaggi)
- Caratteristiche
 - disegno vettoriale/raster
 - image processing
 - interazione mouse/tastiera
 - network communication
 - programmazione OO
- Librerie aggiuntive
 - generazione di suoni e da file
 - send/receive di dati
 - import/export file 2D e 3D



```
sketch_01_intro | Processing 3.5.4
sketch_01_intro
1 import processing.sound.*; // Libreria per la gestione dell'audio
2
3 boolean trovata = false;
4 SoundFile pong; // variabile per l'importazione dei file audio
5 PImage planisphere_base, planisphere_ita; // Le immagini del planisfero base e con l'Italia evidenziata
6
7 void setup() {
8   size(640, 360); // Dimensione della display window in pixel
9   pong = new SoundFile(this, "pong.mp3"); // carica suono di un file audio contenuto in /data
10  planisphere_base = loadImage("Planisphere_640x360.png"); // carica immagine planisfero base in /data
11  planisphere_ita = loadImage("Planisphere+Ita_640x360.png"); // carica immagine planisfero+Ita
12  image(planisphere_base, 0, 0); // visualizza immagine base
13  textAlign(CENTER); fill(0); text("Trova l'Italia!", width/2, height/2); // formattazione e scrittura del testo
14 }
15
16 void draw () {
17 }
18
19 void mouseClicked() { // al click del mouse
20   if (mouseX>298 && mouseX<319 && mouseY>101 && mouseY<126 && !trovata) { // se le coordinate sono quelle dell'Italia
21     trovata=true;
22     image(planisphere_ita, 0, 0); // visualizza immagine con Italia evidenziata
23     pong.play(); // e accompagna con un suono
24     textAlign(CENTER); fill(0); text("Click per giocare ancora!", width/2, height/2); // formattazione e scrittura del testo
25   } else { // altrimenti
26     trovata=false;
27     image(planisphere_base, 0, 0); // visualizza immagine base
28     textAlign(CENTER); fill(0); text("Trova l'Italia!", width/2, height/2); // formattazione e scrittura del testo
29   }
30 }
```

Motivazioni per Processing: Software come mezzo espressivo

- SW (qualità uniche):
 - forme dinamiche e definizione comportamenti
 - elaborazione/analisi gesti e interaction
 - simulazione sistemi naturali e artificiali
 - coordinazione/integrazione multimedia
- Linguaggio come materiale di lavoro
- Sketchbook per lo sviluppo di idee creative
- Programmazione non tipica per informatici (come Logo, Max, ...)

Conoscenza del software

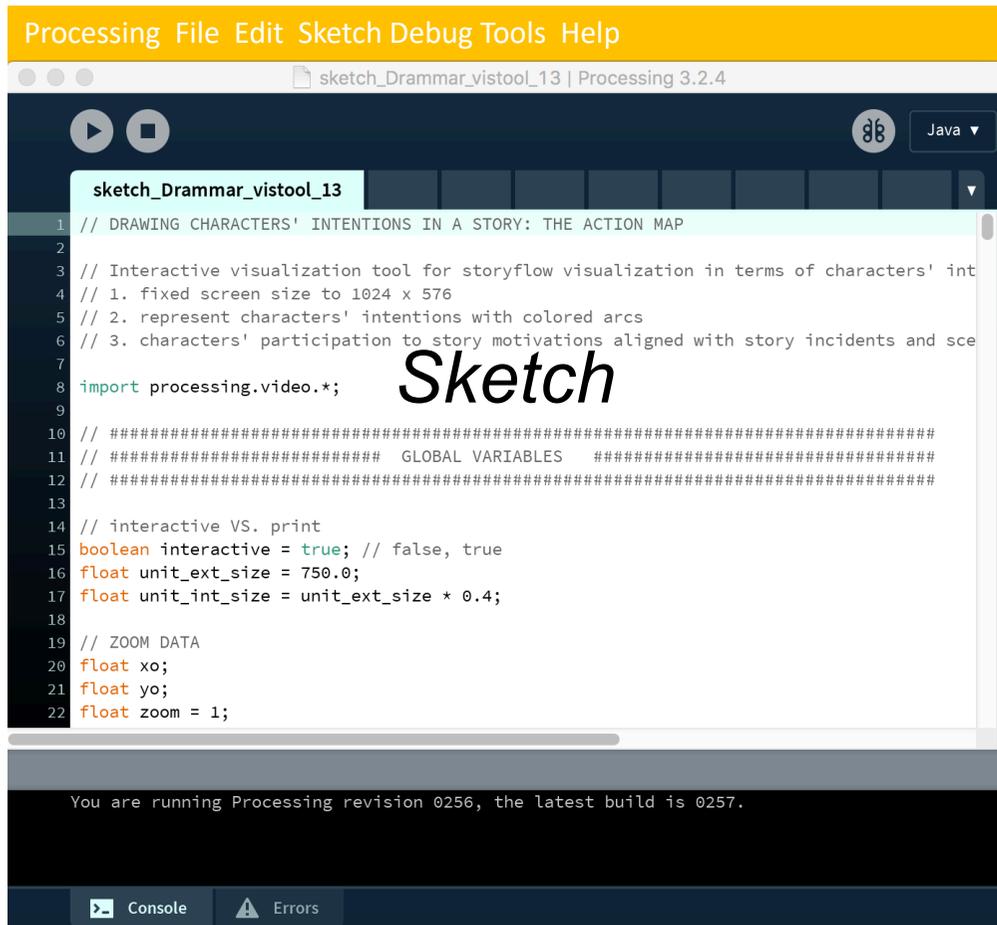
- Conoscenza del software utile per l'espressione (nuove potenzialità): arte "arcana" della programmazione
- Occorre essere "software literate"
- Proprietà di chi possiede la conoscenza:
 - Read: Accedere a materiali e tool creati da altri
 - Write: Creare tool e materiali per altri
- Software: processi che simulano e decidono

Open source

- Poco usato nei software artistici
- Adobe leader di mercato

Ambiente di Processing

Ambiente di Processing 3



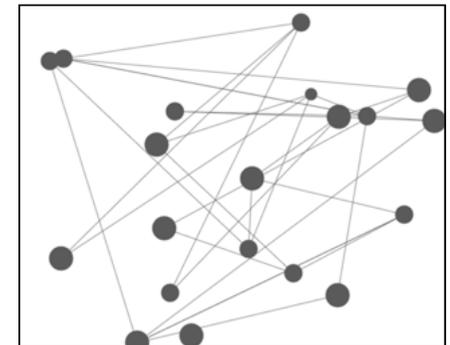
- Menu
- Toolbar
run/stop, new sketch, open, save, export
- Tabs gestione file

- Text editor
Sketch

- Message area
Feedback save/export, display di errori
- Console
output di testo (es. print() e println())

Mode

**Java
Python**



Display window

BOTTONI TOOLBAR

- Run: Compila, apre display window, and runs
- Stop: Termina, non chiude la display window
- Debug: Debuggare lo sketch
 - Step
 - Continue
- Mode: default Java (possibilità di aggiungere modalità)



Comandi da MENU

Processing File Edit Sketch Debug Tools Help

- File: Comandi per gestire e esportare file
- Edit: Controlli per il text editor (Undo, Redo, Cut, Copy, Paste, Find, Replace, etc.)
- Sketch: Run/stop dei programmi, aggiungere media file e librerie di codice
- Debug: Step, Continue, ... Toggle breakpoint
- Tools: Assistenza nell'uso di Processing (automated code formatting, crea fonts, etc.)
- Help: Reference files per linguaggio e ambiente

SKETCH

Sketch

- Tutti i progetti di Processing sono SKETCH
- Ogni SKETCH ha la sua cartella
- Il file principale di uno sketch ha lo stesso nome della cartella e ci sta dentro

- Esempio
 - *Sketch Sketch_123*
 - *Cartella Sketch_123*
 - *File principale nella cartella Sketch_123.pde*
 - *PDE: Processing Development Environment.*

Programma strutturato

- Funzioni `setup()` e `draw()` (max 1+1): struttura studiata per animazione e interattività
- Le variabili dichiarate fuori sono globali
- Esecuzione
 1. Gira il codice fuori da `setup()` e `draw()`
 2. Gira una volta il codice di `setup()`
 3. Gira in loop il codice di `draw()`
 4. Alla fine di ogni loop, si disegna un frame nella DISPLAY WINDOW

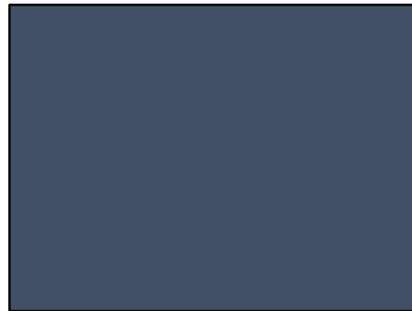
Computer screen

- Dimensioni e tipo di superficie
- Griglia di pixel (molte dimensioni e risoluzioni)
 - HD: 2 Megapixel (1920 wide * 1080 high),
 - HD-ready: 1,310,720 pixels (1280 w * 1024 h)
 - Vecchi display: 786,432 pixels (1024 w * 768 h).
- Risoluzione:
 - Schermi: circa 100 dpi
 - Stampanti: circa 1000 dpi
- Carta fissa, schermi cambiano immagine molte volte al secondo

Display window *size(width, height)*



120x200
size(120, 200)

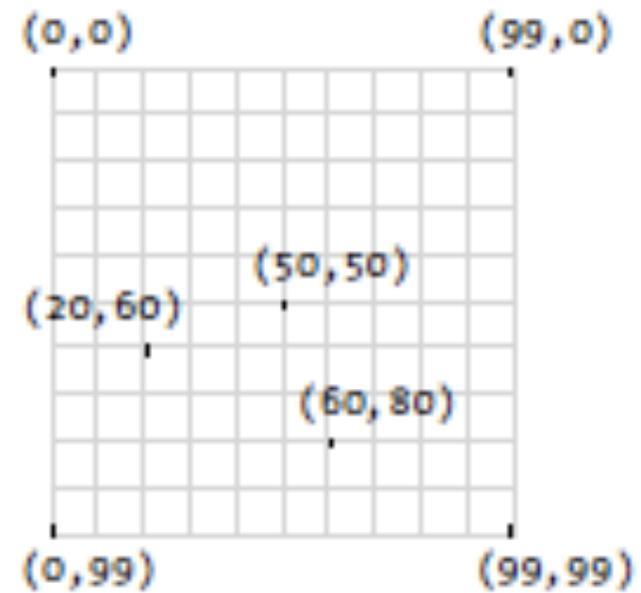
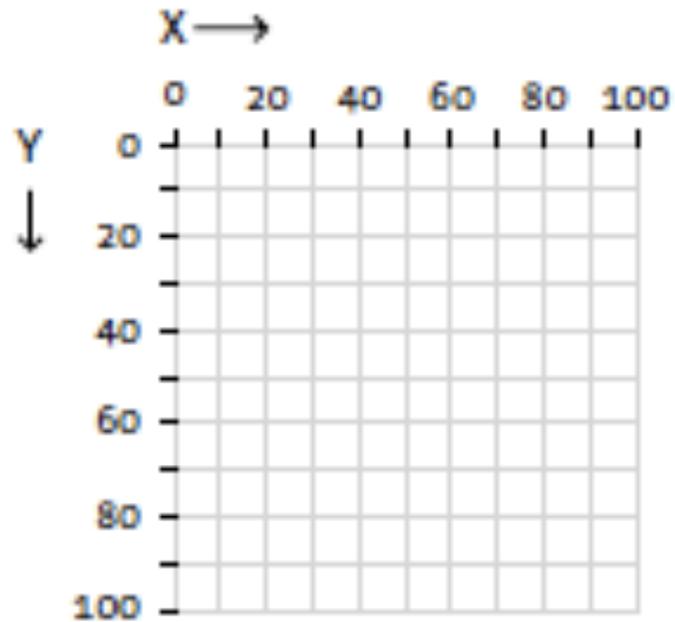


320x240
size(320,240)



240x240
size(240,240)

Display window *coordinate*



Introduzione alla progettazione e programmazione multimediale

Per comprendere tutto il processo produttivo. Scaricare programmi da

<https://www.processing.org>

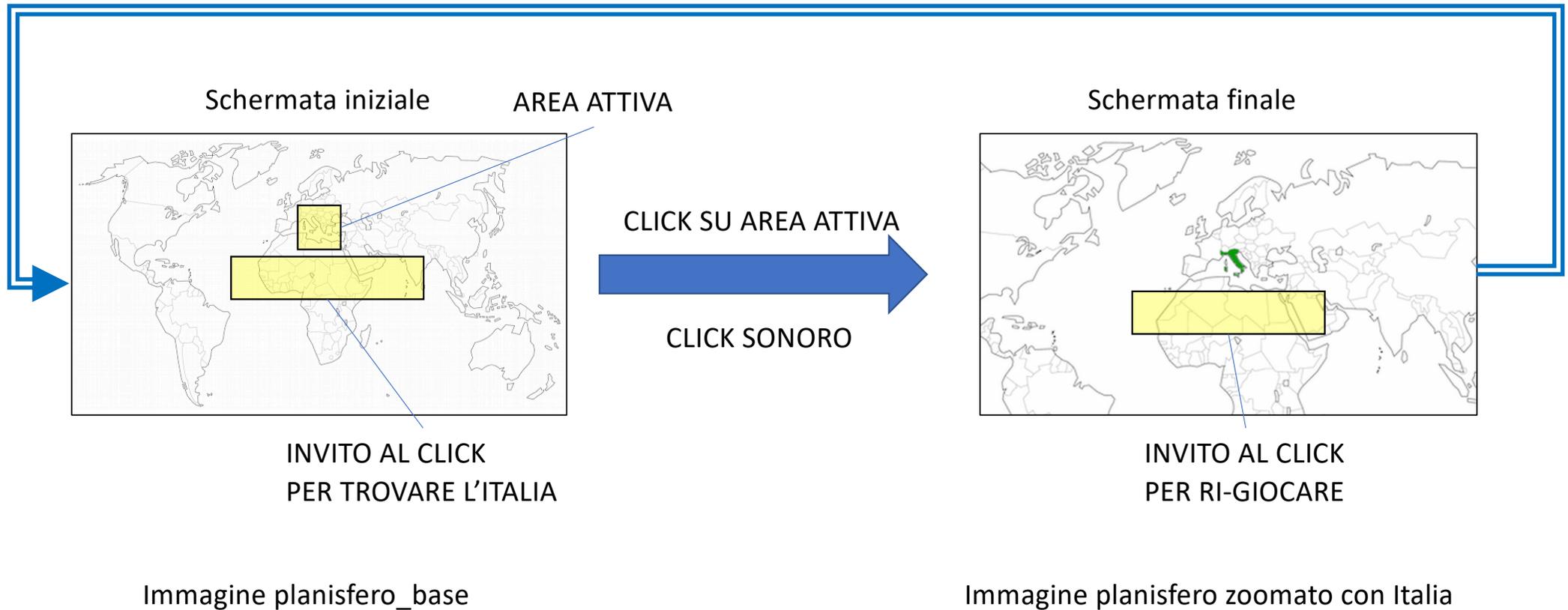
<https://www.audacityteam.org>

<https://www.gimp.org>

Progettazione

- Obiettivo
 - Semplice gioco interattivo
 - Scoprire la posizione dell'Italia sul planisfero
- Metodo
 - Visualizzare un planisfero anonimo
 - Invitare al click nella posizione occupata dall'Italia
 - Se posizione corretta, si visualizza di nuovo il planisfero con l'Italia evidenziata
 - Si può tornare all'inizio per ripetere il gioco

Storyboard



Progetto esecutivo e media file

- Dimensione dello schermo 640x360
- Un'immagine per il planisfero base 640x360
- Un'immagine per il planisfero con Italia evidenziata 640x360
- Un suono per il click, istantaneo, un «click» → idea di ping pong hit
- Mouse-click come interazione
- 2 stati del gioco: «italia da trovare», «italia trovata»

Creazione/recupero dei media

Planisfero base: da Wikimedia commons, licenza CC 2.0-BY



Gentile concessione di
Marguerite Hoedelman

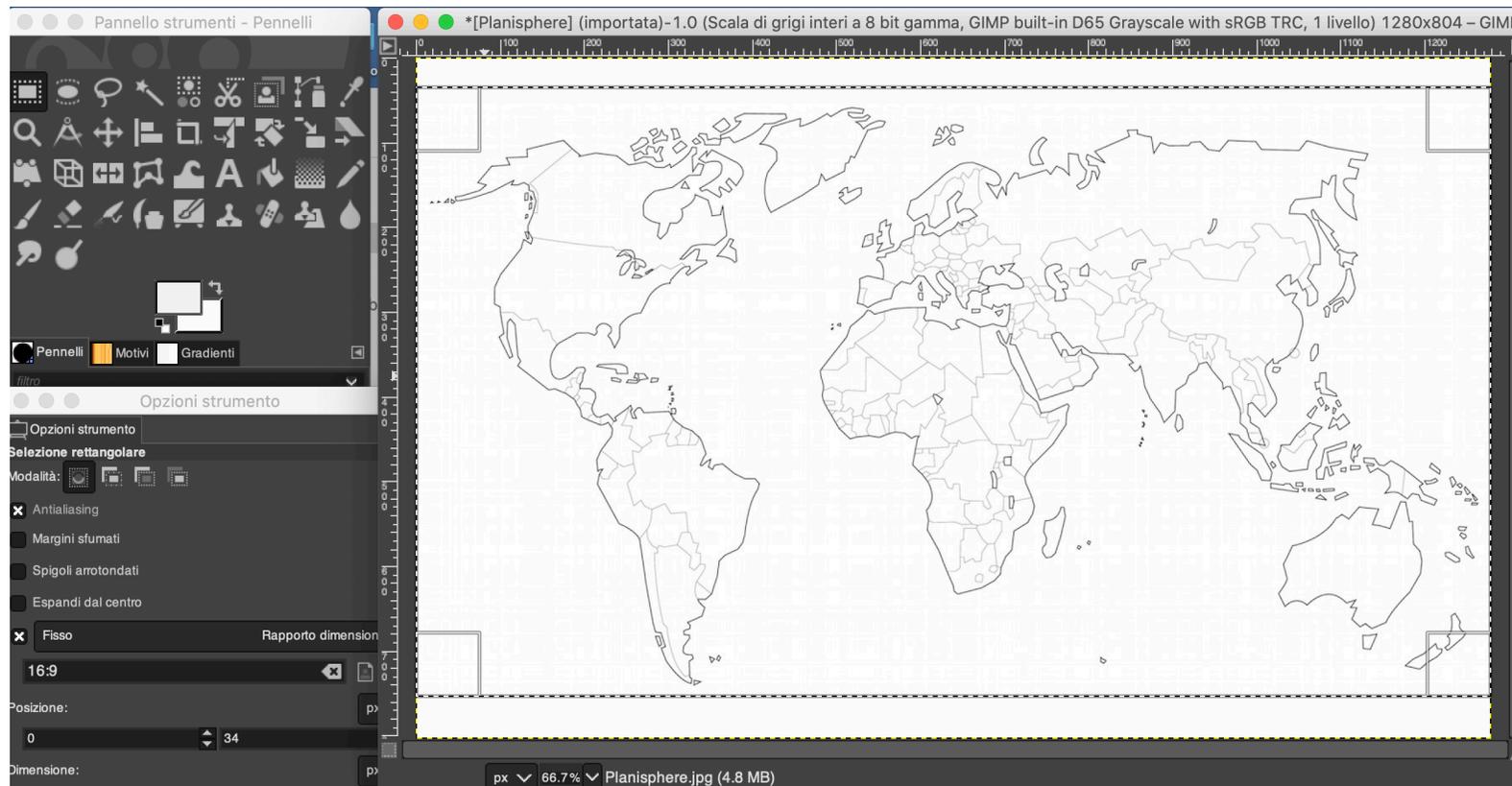
1280x804

Vedere licenze
Creative Commons

Risultato «Planisphere.jpg» su <http://www.di.unito.it/~vincenzo/TecDigSuolmm12/Immagini/>

Elaborazione dei media

Planisfero base: ritaglio a 1280x720 (aspect ratio 16/9), riduzione di scala a 640x360)

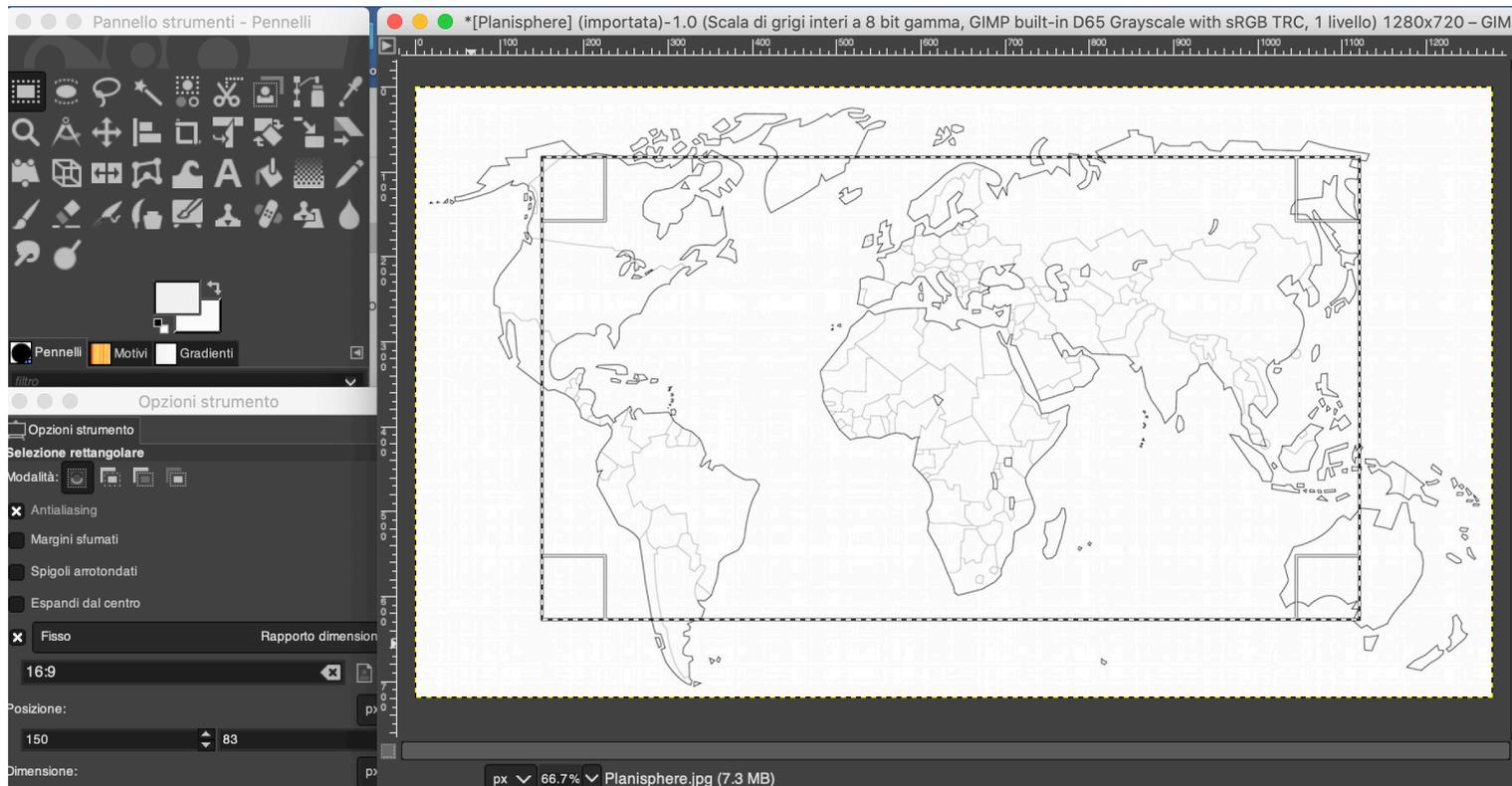


1280x720

Risultato «Planisphere_640x360.jpg» su <http://www.di.unito.it/~vincenzo/TecDigSuolmm12/Immagini/>

Elaborazione dei media

Planisfero zoomato + Italia: selezione riquadro (sempre aspect ratio 16/9) + scala 640x360



Elaborazione dei media

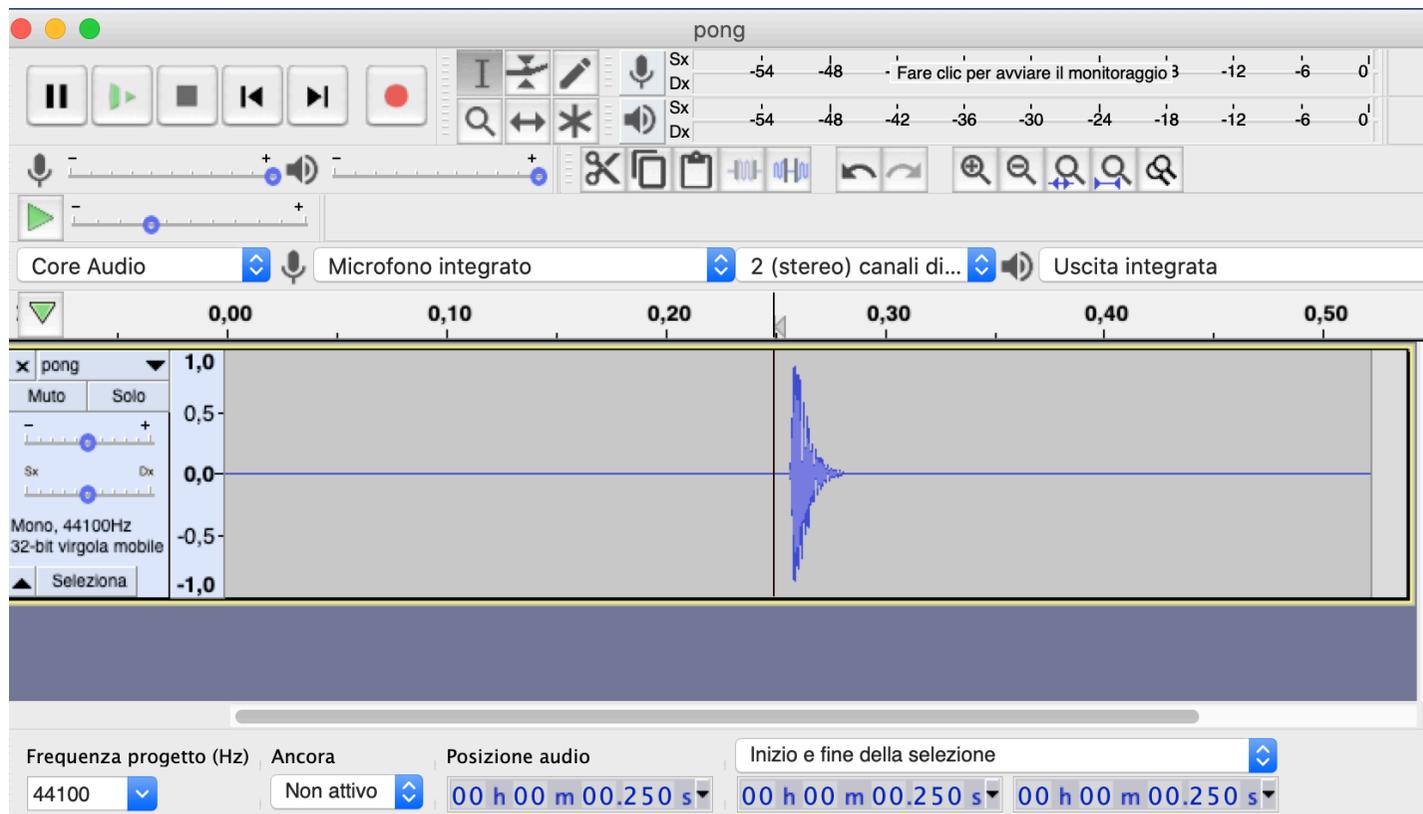
Planisfero zoomato + Italia: Selezione con fuzzy dell'Italia sulla mappa e colorazione



Risultato «Planisphere_Ita_640x360.jpg» su <http://www.di.unito.it/~vincenzo/TecDigSuolmm12/Immagini/>

Creazione/recupero dei media

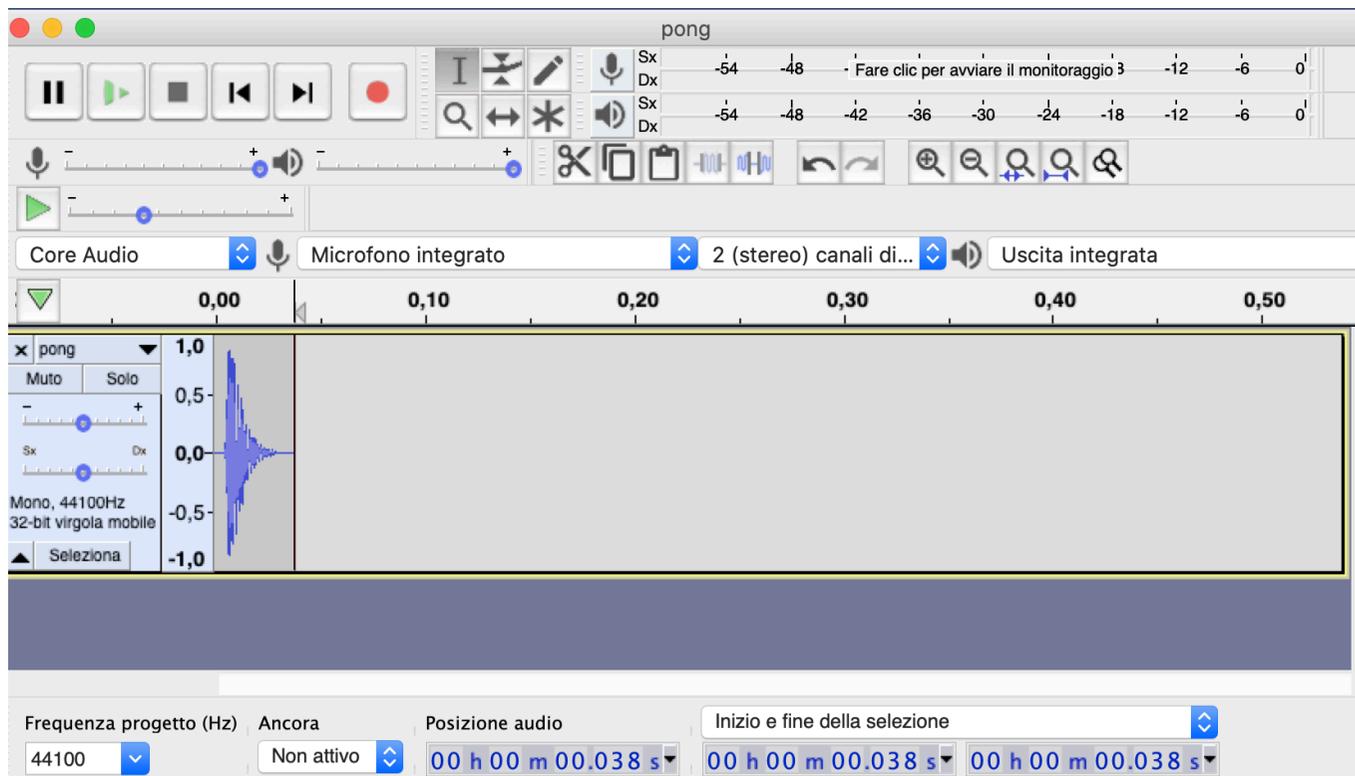
Suono: da freeSound.org, licenza CC 0, public domain



File «pong_orig.mp3» su <http://www.di.unito.it/~vincenzo/TecDigSuolmm12/Suoni/>

Elaborazione dei media

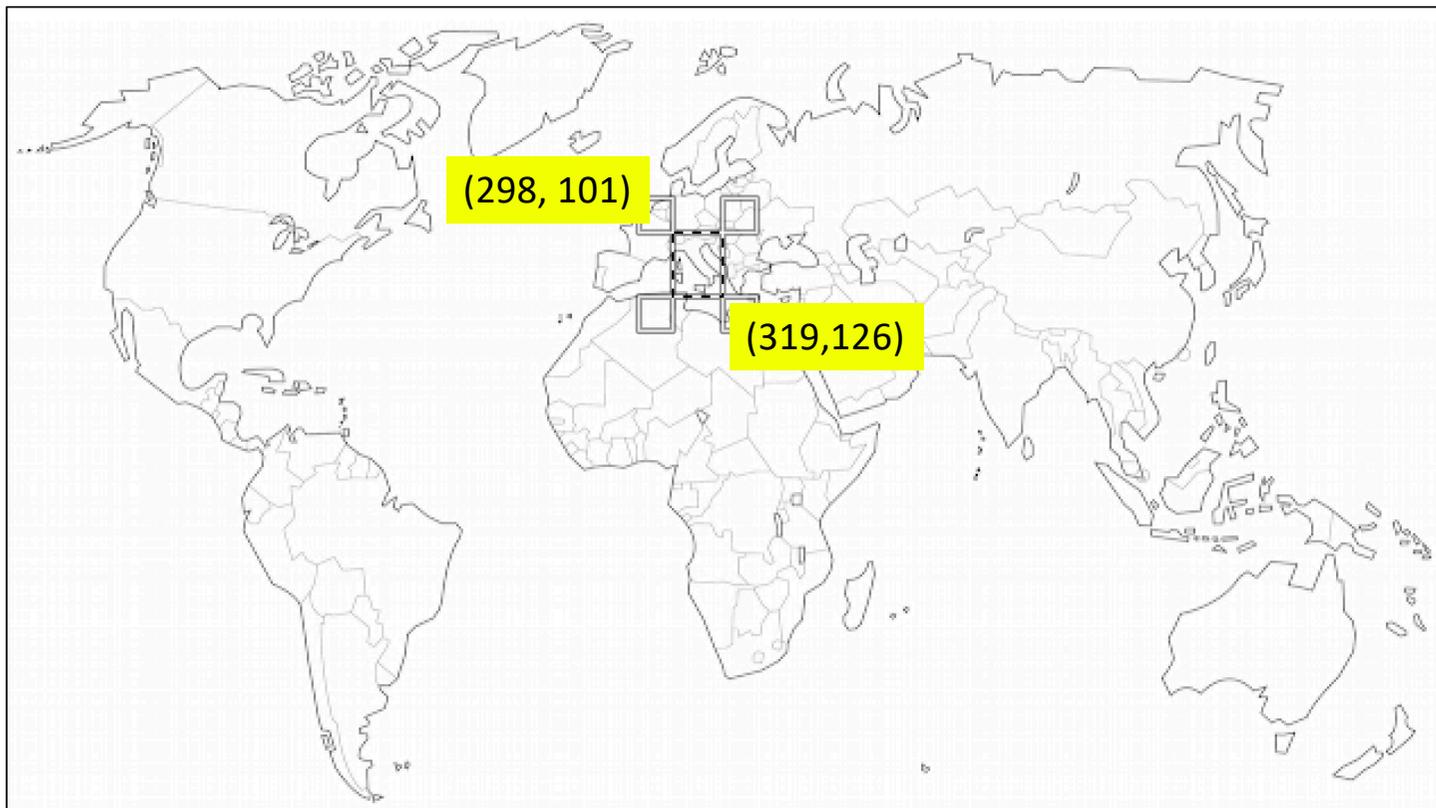
Suono: editing



File «pong.mp3» su <http://www.di.unito.it/~vincenzo/TecDigSuolmm12/Suoni/>

Progettazione dell'interazione

In Gimp, trovare le coordinate del rettangolo comprendente l'Italia



Due stati:

- Trovato
- Non trovato



boolean

Creazione dello sketch

```
sketch_Intro
// import processing.sound.*; // libreria per la gestione dell'audio (SOUND)
import ddf.minim.*; // libreria per la gestione dell'audio (MINIM)

//SoundFile pong; // oggetto suono registrato per click (SOUND)
Minim pong_minim; // oggetto gestore suono (MINIM)
AudioPlayer player; // oggetto player (MINIM)
boolean trovata = false; // trovata l'Italia sulla mappa
PImage planisfero_base, planisfero_ita; // import delle due immagini

void setup() {
  // CARICAMENTO MEDIA
  planisfero_base = loadImage("Planisphere_640x360.jpg"); // immagine base = Italia da trovare trovata (false)
  planisfero_ita = loadImage("Planisphere_Ita_640x360.jpg"); // immagine modificata = Italia trovata (true)
  // pong = new SoundFile(this, "pong.mp3"); // carica file sonoro (SOUND)
  pong_minim = new Minim(this); // crea gestore suono (MINIM)
  player = pong_minim.loadFile("pong.mp3"); // carica file sonoro nel player (MINIM)

  // INIZIALIZZAZIONE DISPLAY
  size(640,360); // Dimensione della display window (aspect ratio 16/9)
  image(planisfero_base,0,0); // visualizza immagine base
  fill(0); textAlign(CENTER); text("Trova l'Italia!", width/2, height/2); // scrivi testo che invita click
}

void draw() {
  void mouseClicked() { // Quando click ...
    if (!trovata && mouseX>298 && mouseX<318 && mouseY>102 && mouseY<125) { // se non ancora trovata Italia e click nel riquadro corretto
      trovata = true; // cambio stato: da 0 a 1
      image(planisfero_ita,0,0); // visualizza immagine modificata
      // pong.play(); // play di suono (SOUND)
      player.rewind(); player.play(); // play di suono (MINIM)
      fill(0); textAlign(CENTER); text("Click per giocare ancora!", width/2, height/2); // scrivi testo che invita click
    } else { // altrimenti (se già trovata)
      trovata = false; // cambio stato: da 1 a 0
      image(planisfero_base,0,0); // visualizza immagine base
      fill(0); textAlign(CENTER); text("Trova l'Italia!", width/2, height/2); // scrivi testo che invita click
    }
  }
}
```

Elementi

Caricamento media

Layout display

Interazione

In esecuzione



Esercizio

- Nuovo funzionamento:
 - a ogni (ri)-avvio, scegliere in modo casuale un paese da identificare
- Progettazione
 - Per ogni paese, preparare la nuove mappe e trovare le coordinate per l'interazione
 - Selezione casuale della proposta
- Implementazione
 - Funzione random() di casualità
 - Array di coordinate e mappe (e suoni?)

Grazie dell'attenzione