CSC103 How Computers Work

Week 7 — Fall 2017

Dominique Thiébaut
dthiebaut@smith.edu
Important Review

- Does the animation leave a trace?
- Are the moving objects move without a trace?
- Are objects appearing in fixed position?
- Learn how to guesstimate positions in the window (see next slide)
Guess where the blue squares are located…
Exercise

replace the circles by squares
Mix circles and squares. Circles only move horizontally with the mouse.
Exercise

Circles move horizontally with the mouse, and squares move vertically.
Using Random Numbers
The random() function

- [https://processing.org/reference/random_.html](https://processing.org/reference/random_.html)

**Description**
Generates random numbers. Each time the `random()` function is called, it returns an unexpected value within the specified range. If only one parameter is passed to the function, it will return a float between zero and the value of the high parameter. For example, `random(5)` returns values between 0 and 5 (starting at zero, and up to, but not including, 5).

If two parameters are specified, the function will return a float with a value between the two values. For example, `random(-5, 10.2)` returns values starting at -5 and up to (but not including) 10.2. To convert a floating-point random number to an integer, use the `int()` function.

**Syntax**
- `random(high)`
- `random(low, high)`
Printing `random()`
Exercise

Replace mouseX and mouseY by calls to random(500)…
Understanding Colors
Understanding Colors
Understanding Colors
Understanding Colors
Understanding Colors
Understanding Colors

Pixel: 23, 13, 19
Understanding Colors

Pixel

23 red
13 green
19 blue
RGB System

• Each pixel corresponds to 3 numbers

• 1st number = shade of red in the color of the pixel

• 2nd number = shade of green in the color of the pixel

• 3rd number = shade of blue in the color of the pixel

• Red-Green-Blue = RGB

• Each shade is represented by 8 bits
• 0,0,0 is black
• 255,255,255 is white
• 100,100,100 is grey
• and so is 200,200,200
• and so is 50,50,50…

• How many shades of grey?
## RGB System

<table>
<thead>
<tr>
<th>1 bit</th>
<th>4 bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0000</td>
</tr>
<tr>
<td>1</td>
<td>0001</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>3 bits</th>
<th>8 bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>0000000</td>
</tr>
<tr>
<td>001</td>
<td>0000001</td>
</tr>
<tr>
<td>010</td>
<td>0000010</td>
</tr>
<tr>
<td>011</td>
<td>0000011</td>
</tr>
<tr>
<td>100</td>
<td>...</td>
</tr>
<tr>
<td>101</td>
<td>1111100</td>
</tr>
<tr>
<td>110</td>
<td>1111101</td>
</tr>
<tr>
<td>111</td>
<td>1111110</td>
</tr>
<tr>
<td>111</td>
<td>1111111</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>2 bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
</tr>
<tr>
<td>01</td>
</tr>
<tr>
<td>10</td>
</tr>
<tr>
<td>11</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>4 bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000</td>
</tr>
<tr>
<td>0001</td>
</tr>
<tr>
<td>0010</td>
</tr>
<tr>
<td>0011</td>
</tr>
<tr>
<td>0100</td>
</tr>
<tr>
<td>0101</td>
</tr>
<tr>
<td>0110</td>
</tr>
<tr>
<td>0111</td>
</tr>
<tr>
<td>1000</td>
</tr>
<tr>
<td>1001</td>
</tr>
<tr>
<td>1010</td>
</tr>
<tr>
<td>1011</td>
</tr>
<tr>
<td>1100</td>
</tr>
<tr>
<td>1101</td>
</tr>
<tr>
<td>1110</td>
</tr>
<tr>
<td>1111</td>
</tr>
</tbody>
</table>
Color Selector
Hue, Saturation, Brightness

Color Selector

D. Thiebaut, Computer Science, Smith College
Good Reference on Color

- Dan Shiffman's page:
  https://processing.org/tutorials/color/
https://www.reddit.com/r/dataisbeautiful/comments/7584no/3d_rgb_scatterplots_of_colours_used_in_famous/
Example

Make the circles blue, and the squares red.
Exercise

Emulate this graphic output

```java
void setup() {
    size(500, 500);
    smooth();
}

void draw() {
}
```
Exercise

Emulate this graphic output

Note that some ellipses are flat, others are tall and skinny...
Variables
Variables

• They contain data

• **Real** numbers, **integer** numbers, **characters**, or Strings

• Variables have a name

• Names always start with a letter or an underscore, and can contain letters or digits: \(x, y, i, alpha, r2d2, c3pO, x1, pi\ldots\)
Variables

• CamelCase

https://upload.wikimedia.org/wikipedia/commons/thumb/e/ef/CamelCase.svg/1200px-CamelCase.svg.png
Example

void setup() {
  size(500, 500);
  smooth();
}

void draw() {
  float radius = random(180);
  float red = random(255);
  float green = random(255);
  float blue = random(255);
  fill(red, green, blue);
  ellipse(random(500), random(500), radius, radius);
}
float x = 3.5;
float y = 5.2;
float sum;

sum = x + y;
println( "sum = " + sum );
Variable Types

- float
- int
- String
  (see https://processing.org/reference/String.html)
- boolean
// define geometry of rectangle
float w = random(100, 300);
float h = random(100, 300);
float x = random(500);
float y = random(500);

// draw a transparent grey version of it
// at 5,5, offset from mouse cursor
noStroke();
fill(200, 200, 200, 100);
rect(x+5, y+5, w, h);

// draw a randomly colored rectangle at mouse cursor
float red = random(255);
float green = random(255);
float blue = random(255);
fill(red, green, blue);
rect(x, y, w, h);
We stopped here last time...
Loops

(See https://processing.org/reference/for.html)
for ( int i=0;  i <= 10;  i=i+1 ) {
    println( i );
}
Example 1

```java
for ( int i=0; i <= 10; i=i+1 ) {
    println( i );
}
```

i++
Example 1

```java
for ( int i=0; i <= 10; i++ ) {
    println( i );
}
```
Example 2

```java
int sum = 0;
for ( int counter=0; counter <= 10; counter++ ) {
    sum = sum + counter;
}
println( "sum = ", sum );
```
```java
void draw() {
    fill( 200, 100, 0 );

    for ( int x=50; x<500; x = x + 100 ) {
        ellipse( x, 100, 40, 40 );
    }
}
```
Example 4

```java
void draw() {
    fill( 200, 100, 0 );

    for ( int y=50; y<500; y = y + 100 ) {
        ellipse( 100, y, 40, 40 );
    }
}
```
Example 5

```cpp
void draw() {
    fill( 200, 100, 0 );
    for ( int x=50; x<500; x = x + 100 ) {
        for ( int y=50; y<500; y = y + 100 ) {
            ellipse( x, y, 40, 40 );
        }
    }
}
```
The if statement: Testing conditions
Example

Color circles red on left side of window, and blue and right side.

red —> fill(242, 66, 213);
blue —> fill(49, 132, 245);
void draw() {
    if ( ) {
        // set fill color to red
        fill( 242, 66, 213 );
    }
    else {
        // set fill color to blue
        fill( 49, 132, 245 );
    }
    ellipse( mouseX, mouseY, 80, 80 );
}
if (boolean expression) {
    statement;
    statement;
}
else {
    statement;
    statement;
}
The If-Statement

```java
if ( boolean expression ) {
    statement;
    statement;
} else {
    statement;
    statement;
}
```

<table>
<thead>
<tr>
<th>Comparison</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;</td>
<td>xxx &lt; yyy</td>
</tr>
<tr>
<td>&lt;=</td>
<td>xxx &lt;= yyy</td>
</tr>
<tr>
<td>&gt;</td>
<td>xxx &gt; yyy</td>
</tr>
<tr>
<td>&gt;=</td>
<td>xxx &gt;= yyy</td>
</tr>
<tr>
<td>==</td>
<td>xxx == yyy</td>
</tr>
<tr>
<td>!=</td>
<td>xxx != yyy</td>
</tr>
</tbody>
</table>
Example

Draw circles only if the mouse button is pressed…
Animation with Processing
Reference Page

#1: Create the Skeleton Sketch

```cpp
// aquarium0

void setup() {
    size( 1000, 600 );
    smooth();
}

void draw() {
}
```
#2: Create Data Directory
#3a: Find an animated gif fish with transparent background

https://www.google.com/search?q=animated+gif+fish+transparent
#3b: Save the Fish to Data
#4: Find image of a Fish Tank
#5b: Save tank to Data
#6: Display Fish and Tank

```java
PImage fish;
PImage tank;
void setup() {
  size( 900, 500 );
  smooth();
  fish = loadImage( "fish.gif" );
  tank = loadImage( "tank.jpg" );
}
void draw() {
  image( tank, 0, 0, width, height );
  image( fish, 100, 100 );
}
```
#6: Display Fish and Tank

// aquarium0

PImage fish;
PImage tank;

void setup() {
    size( 900, 500 );
    smooth();
    fish = loadImage( "fish.gif" );
    tank = loadImage( "tank.jpg" );
}

draw() {
    image( tank, 0, 0, width, height );
    image( fish, 100, 100 );
}
#7a: Break up Fish into Frames

https://ezgif.com/split
#7b: Store Frames into Data
PImage[] fish = new PImage[8];
PImage tank;

void setup() {
    size( 900, 500 );
    smooth();
    fish[0] = loadImage( "fish0.gif" );
    fish[1] = loadImage( "fish1.gif" );
    fish[2] = loadImage( "fish2.gif" );
    fish[3] = loadImage( "fish3.gif" );
    fish[4] = loadImage( "fish4.gif" );
    fish[5] = loadImage( "fish5.gif" );
    fish[6] = loadImage( "fish6.gif" );
    fish[7] = loadImage( "fish7.gif" );
    tank = loadImage( "tank.jpg" );
}

void draw() {
    image( tank, 0, 0, width, height );
    image( fish[frameCount % 8], 100, 100 );
}
# 9: Make the Fish Move Forward

PImage[] fish = new PImage[8];
PImage tank;
int x = 0;
int y = 100;

void setup() {
    size( 900, 500 );
    smooth();
    fish[0] = loadImage( "fish0.gif" );
    fish[1] = loadImage( "fish1.gif" );
    fish[2] = loadImage( "fish2.gif" );
    fish[3] = loadImage( "fish3.gif" );
    fish[4] = loadImage( "fish4.gif" );
    fish[5] = loadImage( "fish5.gif" );
    fish[6] = loadImage( "fish6.gif" );
    fish[7] = loadImage( "fish7.gif" );
    tank = loadImage( "tank.jpg" );
    frameRate( 10 );
}

void draw() {
    image( tank, 0, 0, width, height );
    image( fish[frameCount % 8], x, y );
    x = x + 5;
    if ( x > width ) {
        x = -300;
    }
}
#10: Fancier Tank