

EUTOPIA 2014 CLONACIÓN, ORIGINAL Y COPIA

CIU CÓ DADDI GO

TONI CASTILLO
PABLO GALLEGO

```

numeroMaximoCasas=120; //Numero de Casas que se crearan
numeroMaximoPersonas=160; //Numero de personas que se crearan. Las que se
anchoBordeMarco=30;

at distanciaMinimaSeparacionCasasBarrio=24.0;
at distanciaMaximaSeparacionCasasBarrio=50.0;
numeroMaximoDeBarrios = 7;

2DProcessing box2d;
asa [] casas = new Casa[numeroMaximoCasas];
ayList<Casa> casas;
ersona [] personas = new Persona[100];
ayList<Persona> personas;
ayList<Borde> bordes;

1 setup() {
//Fon el fondo de presentación en blanco (no el del sketch)
(javax.swing.JFrame) frame).getContentPane().setBackground(new java.awt.Color(
ize(768, 768);
ramaRate(30);

ox2d = new Box2DProcessing(this);
ox2d.createWorld();
ox2d.setGravity(0,0);

ordes = new ArrayList<Borde>();
ordes.add(new Borde(width/2,height,width,anchoBordeMarco*2));
ordes.add(new Borde(width/2,0,width,anchoBordeMarco*2));
ordes.add(new Borde(width,height/2,anchoBordeMarco*2,height));
ordes.add(new Borde(0,height/2,anchoBordeMarco*2,height));

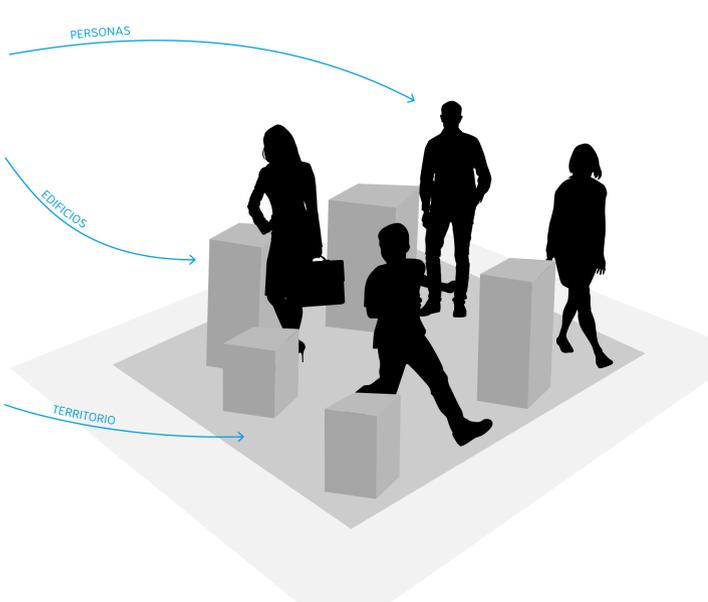
1 draw() {

ackground(255);
//Actualiza el mundo
ox2d.step();
    
```

REPRESENTAR LA CIUDAD

Haremos una interpretación subjetiva de la ciudad con un sistema de representación mediante código y datos –arte generativo–. El sistema debe dar como resultado una imagen, serie de imágenes o animación impredecibles y a la vez controladas. Modificando los datos subyacentes y las variables de cada elemento obtendremos nuevas versiones, copias o clonaciones. Hemos considerado 4 elementos básicos y la relación entre ellos para representar una ciudad.

- Territorio
- Edificios
- Personas
- Tiempo



ELEMENTOS DEL SISTEMA

1 El territorio.

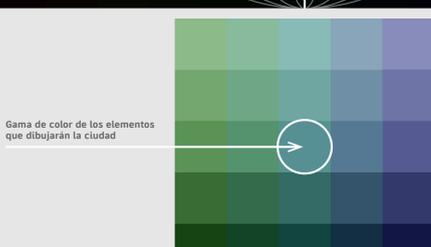
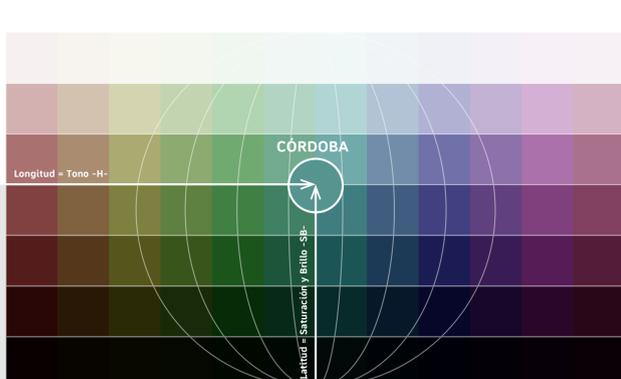
El territorio lo hemos definido con dos elementos: un área delimitada dentro de otro área mayor –en nuestro caso un cuadrado dentro de un cuadrado– y mediante un color o gama de colores.

Para definir el color de nuestra ciudad hemos definido un color para Córdoba interpolando las coordenadas GPS de Córdoba a color HSB.

Los 360° del ecuador son los 360° de tono –H igual a longitud– con el 50% de saturación y 50% de Brillo. Hacia el norte reducimos brillo y aumentamos saturación y hacia el sur a la inversa –S y B igual a latitud– siempre en proporción inversa.

Coord. **GPS** de Córdoba: 37.8833, -4.7666
Color **HSB** de Córdoba: 175, 42, 58

En base a ese color construiremos el resto de los colores de la ciudad tomando como centro el color Córdoba y variando un porcentaje en todos los sentidos. Este porcentaje se obtiene de un cálculo sobre la nubosidad en las coordenadas GPS y en el momento en que se ejecuta. El dato de la nubosidad lo tomamos de un servicio web.

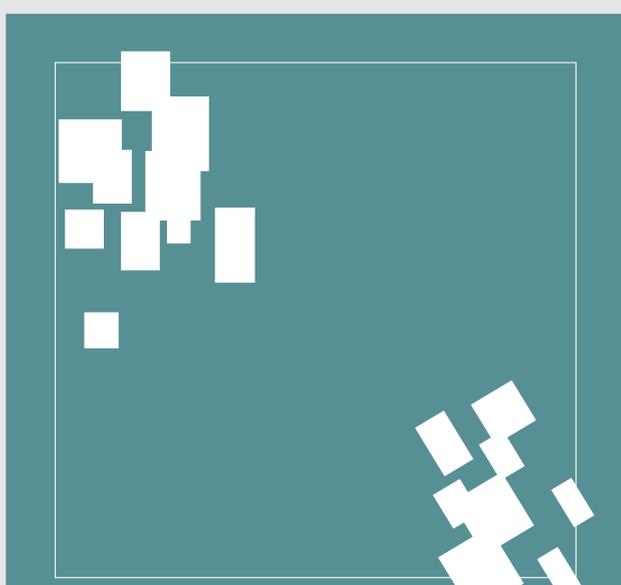


2 Los edificios.

En nuestro modelo los edificios "se construyen" por imitación y proximidad, dentro de unas dimensiones y número limitados.

Esto generará un "tejido urbano" impredecible creando barrios, plazas y calles en los espacios libres que se generen entre los "edificios".

De igual manera los edificios tienen una "vida" y por tanto algunos desaparecerán y aparecerán otros en su lugar.

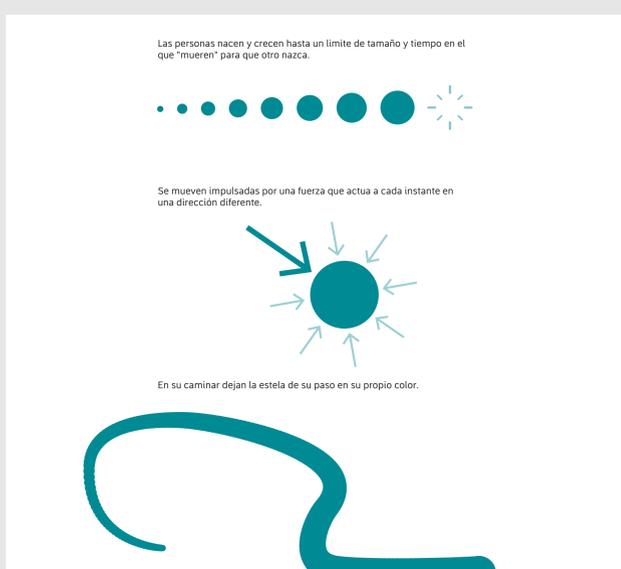


3 Las personas.

Las personas las representaremos mediante círculos que nacen, crecen y mueren. Al nacer adquieren un color de la ciudad salvo algunos que aleatoriamente adquieren un color único –blanco–

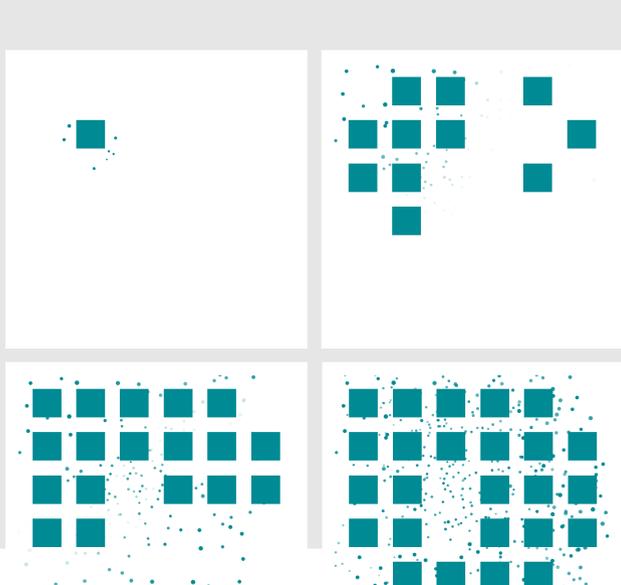
Las personas se mueven por la ciudad impulsadas continuamente en cualquier dirección quedando el rastro de su paso. Su caminar es más lento conforme más "edad" tienen.

Las personas se mueven en los límites de la ciudad y no pueden pasar sobre los edificios ni sobre otras personas.



4 El tiempo.

Edificios y personas crecen en el territorio y a la vez. Sin embargo los edificios son elementos más estables y con un periodo de existencia mucho más largo que las personas, algunos desaparecen y otros aparecen. Las personas viven periodos cortos en relación a los edificios y su acción se superpone y solapa unas con otras de una manera más orgánica.



```

numeroMaximoCasas=120; //numero de Casas que se crearian
numeroMaximoPersonas=160; //Numero de personas que se crearan. Las que se
anchoBordeMarco=30;

at distanciaMinimaSeparacionCasasBarrio=24.0;
at distanciaMaximaSeparacionCasasBarrio=50.0;
numeroMaximoDeBarrios = 7;

2DProcessing box2d;
asa [] casas = new Casa[numeroMaximoCasas];
ayList<Casa> casas;
ersona [] personas = new Persona[100];
ayList<Persona> personas;
ayList<Borde> bordes;

1 setup() {
//Fon el fondo de presentación en blanco (no el del sketch)
(javax.swing.JFrame) frame).getContentPane().setBackground(new java.awt.Color(
ize(768, 768);
ramaRate(30);

ox2d = new Box2DProcessing(this);
ox2d.createWorld();
ox2d.setGravity(0,0);

ordes = new ArrayList<Borde>();
ordes.add(new Borde(width/2,height,width,anchoBordeMarco*2));
ordes.add(new Borde(width/2,0,width,anchoBordeMarco*2));
ordes.add(new Borde(width,height/2,anchoBordeMarco*2,height));
ordes.add(new Borde(0,height/2,anchoBordeMarco*2,height));

1 draw() {

ackground(255);
//Actualiza el mundo
ox2d.step();
    
```