

The background features a network of blue circles of varying sizes connected by thin black lines. These lines form a complex web of polygons, primarily triangles and quadrilaterals, scattered across the white background. The circles are semi-transparent, allowing the lines and other circles to be seen through them.

# Processing

The background is a dark blue/black field filled with numerous semi-transparent blue circles of varying sizes. Overlaid on this is a complex, white network graph consisting of many small nodes connected by thin lines, creating a web-like pattern that fills the entire frame.

# ← Processing

Exhibition.

A curated collection of projects created with Processing.  
New software is added each month.

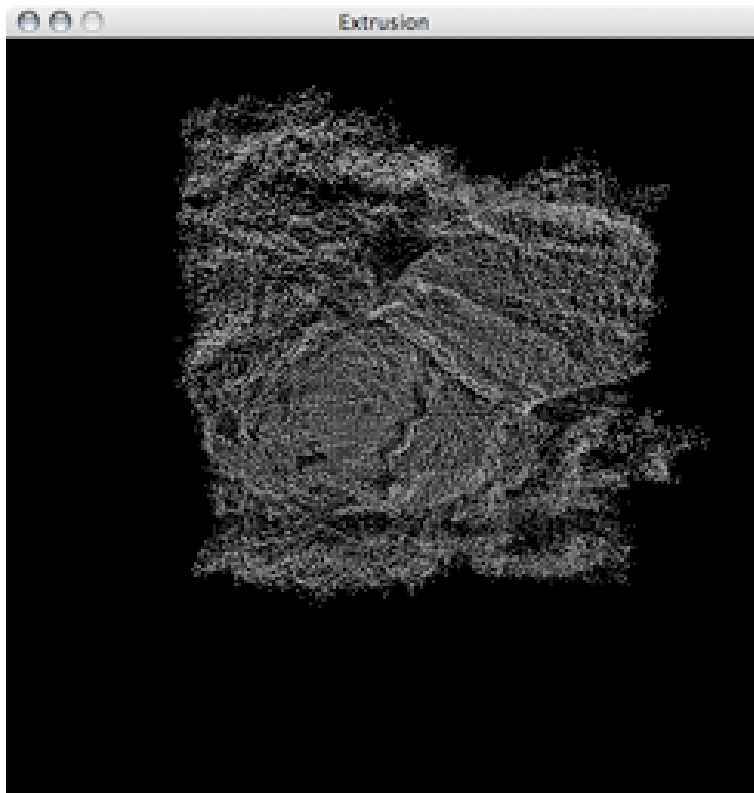
# ¿Por qué Processing?

- Multiplataforma (Win, Mac, Linux)
- Lenguaje abierto
- Diseñado para artistas y diseñadores
- Gran comunidad de usuarios
- Gran N de ejemplos y tutoriales
- Buen N de librerías
- Optimizado para visualizaciones

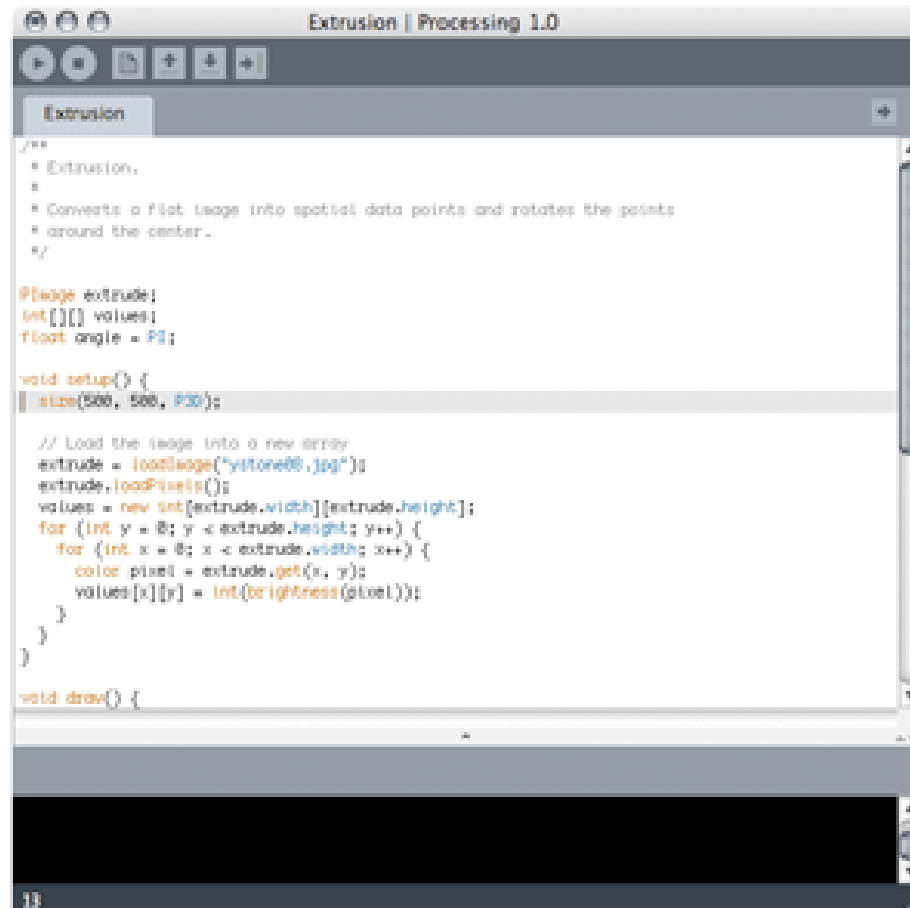
# Processing

- Página web oficial [www.processing.org](http://www.processing.org)
- Descargar versión 2.2.1
- Instalación
- Probamos el programa

# IDE Processing



Display Window



Toolbar

Tabs

Text Editor

Message Area

Text Area

# Introducción a la Programación

- Programar es escribir un conjunto de instrucciones o funciones para que el ordenador las ejecute

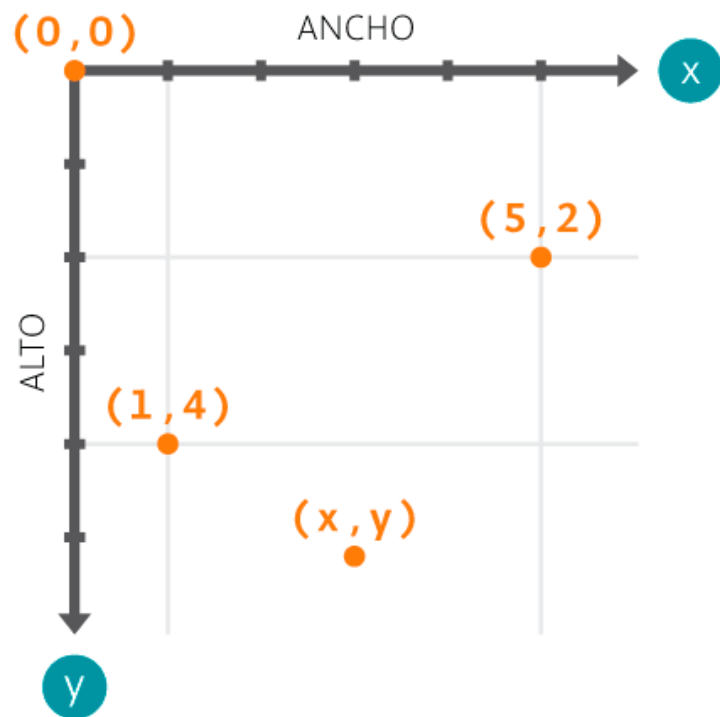
funciones:            `print ( ) ;`

comentarios:        `// comentario`

expresiones:        `2 + 2`

# Pantalla y Pixeles

## Coordenadas



## Funciones

`point(x,y);`

`line(x1,y1,x2,y2);`

`rect(x1,y1,ancho,largo);`

`ellipse(xc,yc,ancho,largo);`

# Colores y estilo

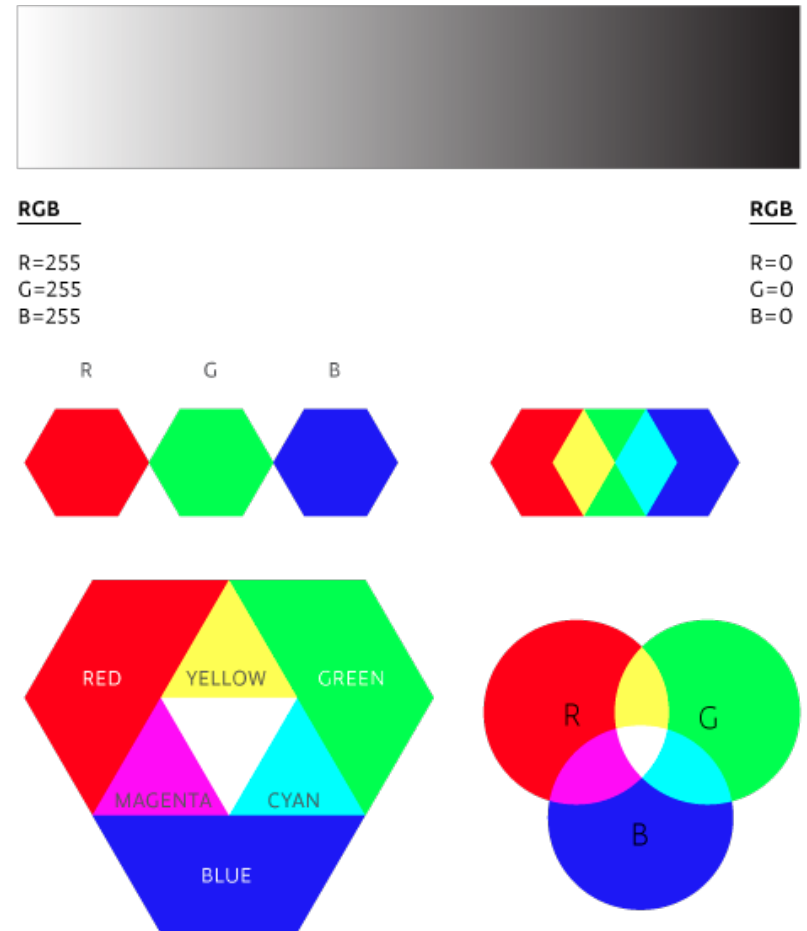
- Cada color se construye con 3 valores (R,G,B) de 0 a 255

## Funciones

```
background(color);  
stroke(color);  
fill(color);
```

```
strokeWeight(grueso);  
noStroke();  
noFill();
```

```
color = gris  
color = rojo, verde, azul  
color = rojo, verde, azul, transparencia
```

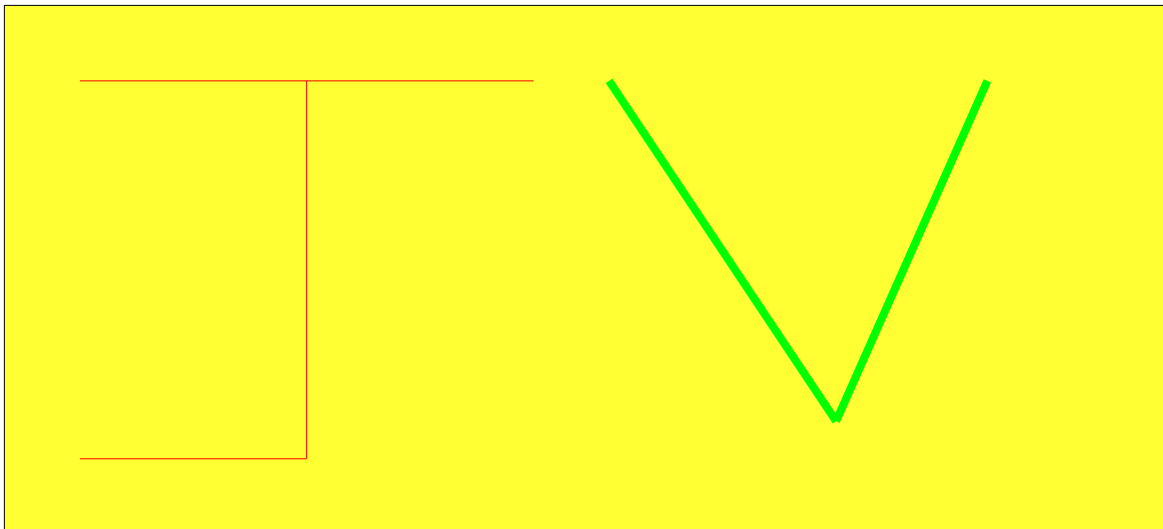




# Primer Ejercicio

Escribe tus iniciales

- A mano podemos hacer un esquema del dibujo y definimos los puntos



# Programación Dinámica

- Para hacer una programación dinámica usamos las funciones `setup()` y `draw()`

```
void setup( ) {
```

```
}
```

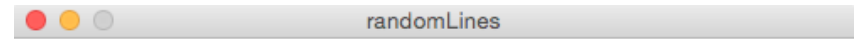
```
void draw( ) {
```

```
}
```

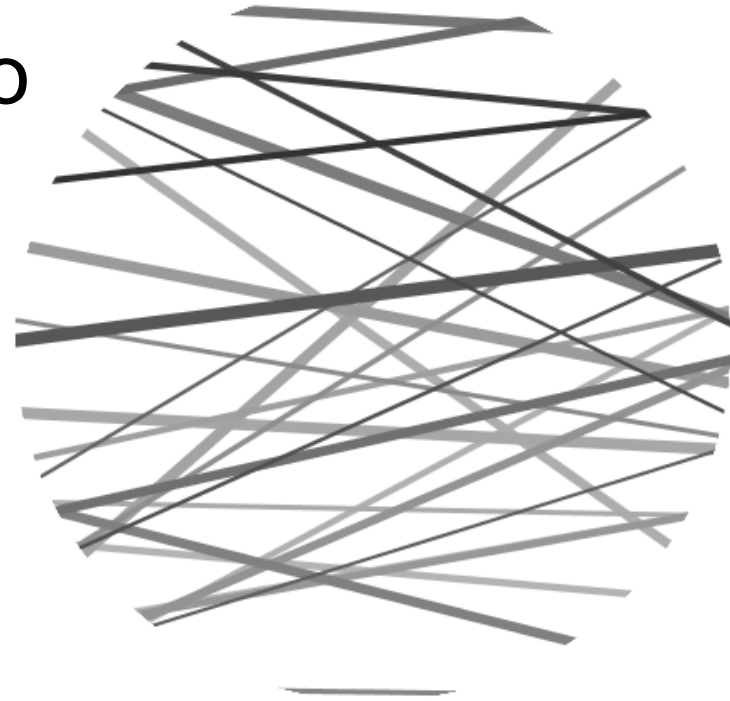
# Variables

- Variable = Nombre simbólico para un dato
- Tipos de datos
  - Int = entero, 1 , 3, -19, 2340
  - Float = número flotante, 3.4, -5.6, 100.09
  - Boolean = true o false
  - Char = 'a' 'b' ' '
  - String = cadena de caracteres, “casa”, “Pedro”, “Caballo”

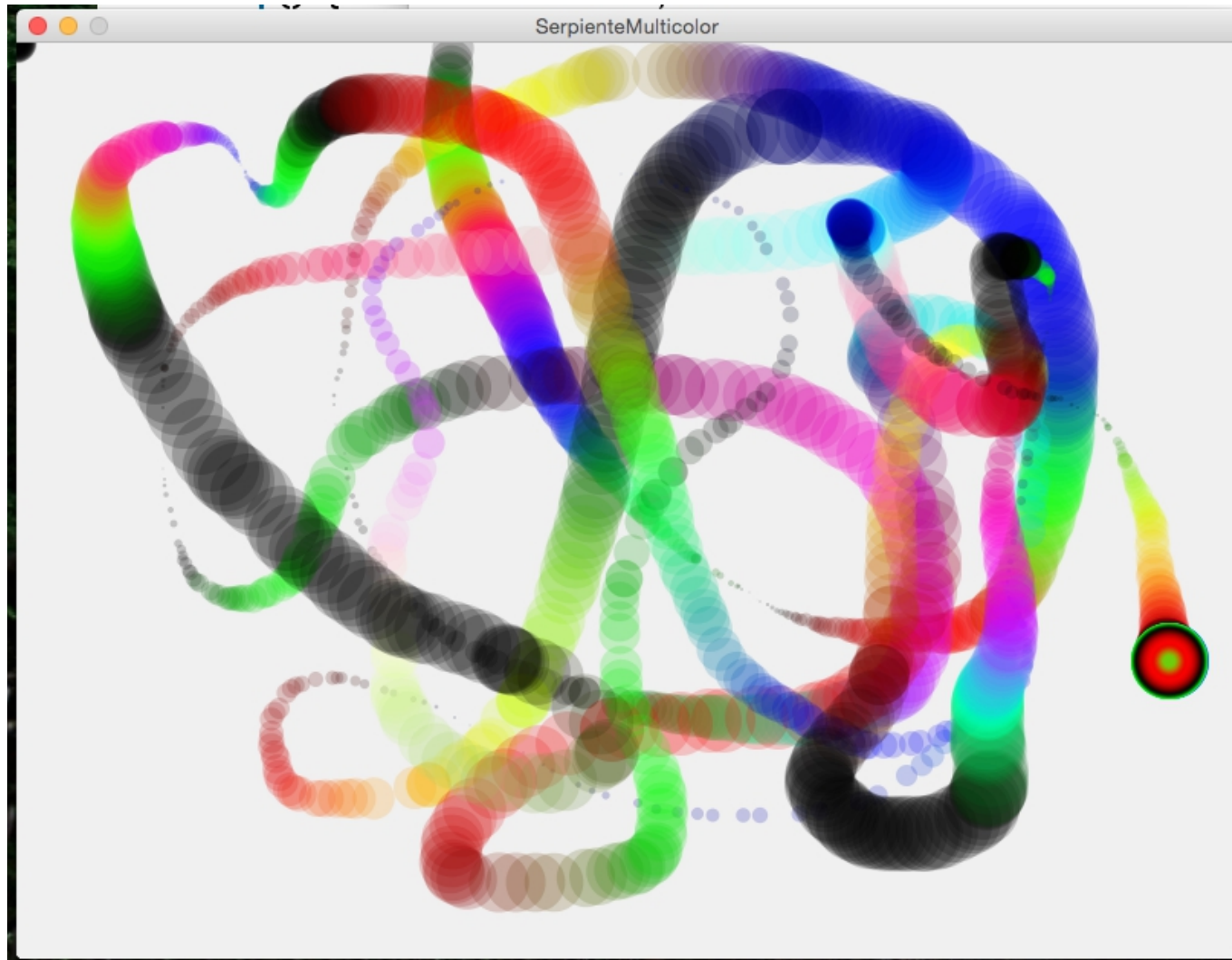
# Ejercicio:random()



- Función random Aleatorio



# Ejercicio: Serpiente Multicolor



# Ejercicio: Reloj de Post-it

- Uso de imagenes

`PImage`

- Arrays

`Int[] vector = new int[];`

- bucle for

```
for(int i=0; i<N; i++){  
}
```

