



## Fundamentos de Programación Orientada a Objetos

Tecnicatura Universitaria en diseño integral de videojuegos

Trabajo Práctico N° 2  
Uso de estructuras  
condicionales y  
repetitivas  
07/04/2021

### Objetivo de aprendizaje:

- Uso de estructuras condicionales
- Uso de variables globales y locales.
- Uso de estructuras de control repetitiva: for, do-while, while

Utilizando Processing realice los siguientes ejercicios.

**Nota:** Por cada ejercicio debe crear un nuevo sketch, con el nombre por ejemplo: sketchEjercicio1

### Ejercicio 1: Utilice setup() y draw()

Dibujar sobre un lienzo de tamaño (500,500) un cuadrado que mantenga una distancia desde el borde superior (eje x) de 20px y una distancia de 20px desde el borde lateral izquierdo (eje y).

El ancho y el alto del rectángulo miden 20px.

Declare las siguientes variables: x, y, anchoRectangulo, altoRectangulo, margenDerechoX. Todas de tipo de dato **int**.

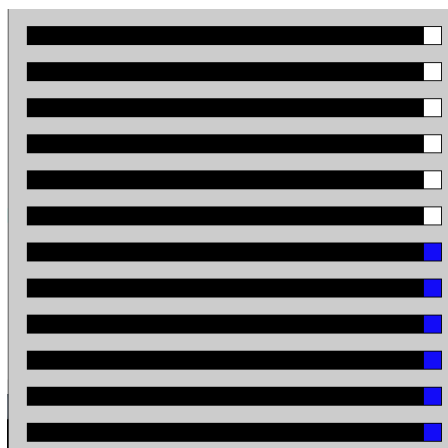
Asigne los valores apropiados a cada una de sus variables, de tal forma que se pueda resolver el ejercicio.

Dentro de la función draw() luego de dibujar el rectángulo, el valor de x se incrementa en una unidad y el valor de y se incrementa en el doble del alto del rectángulo si se cumple la condición de que el valor de x es igual al margenDerechoX menos el ancho del rectángulo, entonces se debe asignar el valor 20 a x.

Luego evaluar el valor de y, si su valor es mayor o igual a la **mitad** de la altura del lienzo, entonces cambiar el color de relleno del rectángulo (el color debe ser azul).

Por último evaluar si **y** es igual o superior a la altura del lienzo (**height**), si es así imprimir por consola los valores de **x** e **y**, luego invocar a la función **noLoop()**

El resultado debería verse similar a la siguiente figura:



## Ejercicio 2: Utilice setup() y draw()

Declare las variables necesarias para dibujar una línea que se dibuja desde las coordenadas iniciales del lienzo y se extiende por todo el ancho.

Sobre el punto medio de la línea y a una distancia de 40px (en sentido vertical desde la línea) dibuje una elipse que tenga como ancho 80px y de alto 80px.

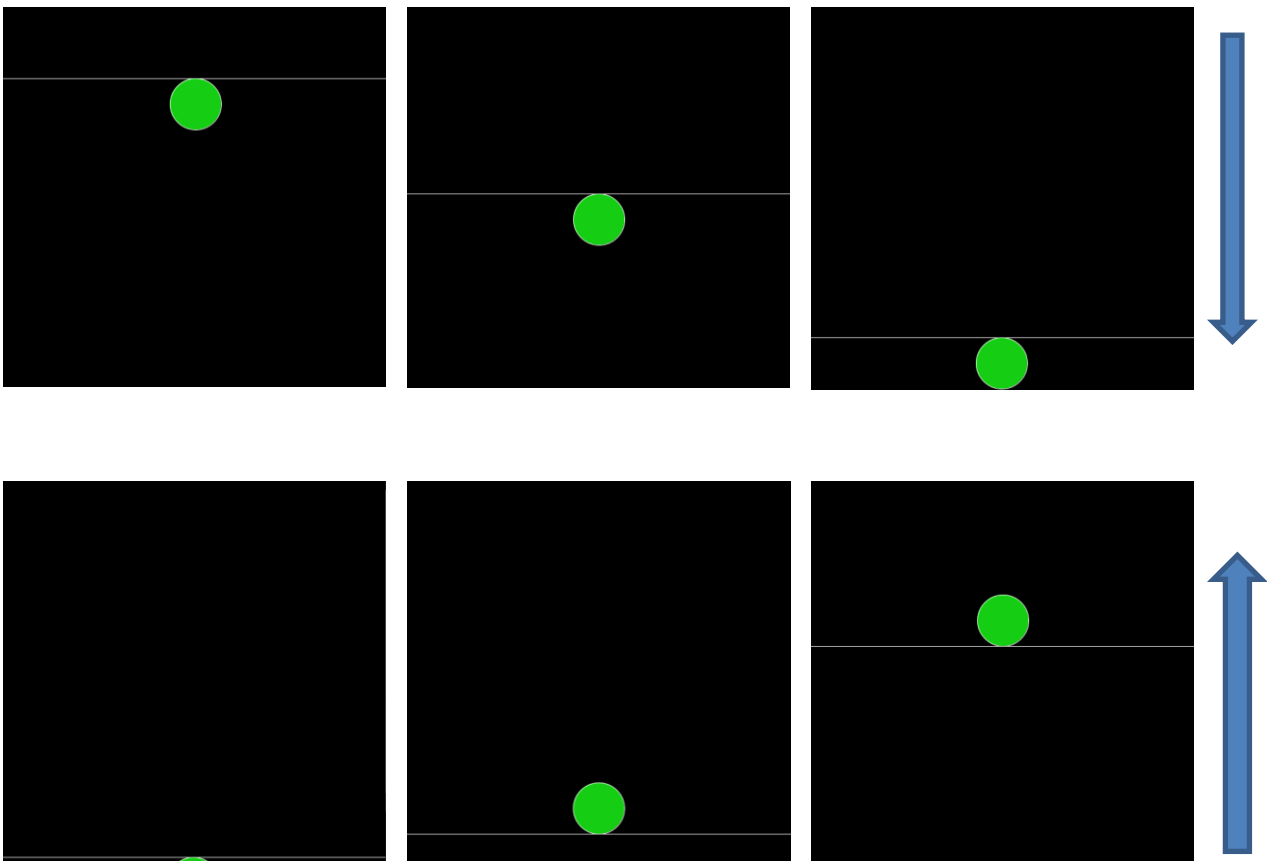
Dentro de la función draw(), actualice las variables necesarias para que la línea desde su inicio se mueva en dirección hacia abajo arrastrando la elipse.

Mantenga en cero el valor para background().

Cuando la línea supere la posición de la altura del lienzo, debe invertir su sentido, es decir dirigirse hacia arriba arrastrando la elipse.

Cuando la línea alcance nuevamente el valor 0 para su posición en y, el desplazamiento debe ser hacia abajo y así sucesivamente.

El lienzo debería verse como en las siguientes figuras:

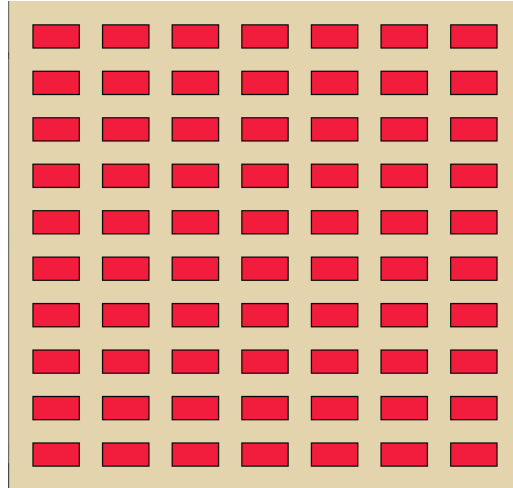


### Ejercicio 3: Utilice setup()

Dibuje en toda la extensión del lienzo de (440, 420) rectángulos de idénticas medidas (40 ancho y 20 de alto) y que mantengan una distancia de 20 pixeles entre ellos tanto horizontal como verticalmente.

Utilice la estructura de control repetitiva **for**.

El lienzo debería verse así:



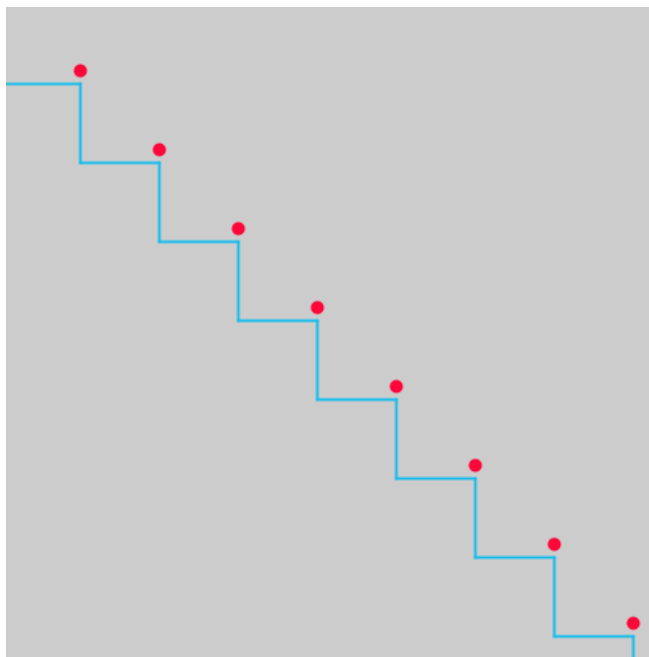
**Ejercicio 4:** Utilizando la estructura de control repetitiva while() dibuje la siguiente imagen utilizando líneas que forman escalones y sobre cada borde de escalón se dibuje un punto de color rojo.

El tamaño del lienzo es size(500,500).

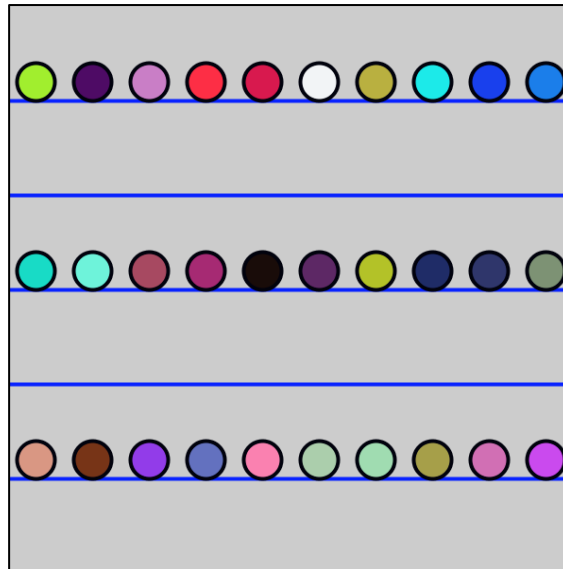
La estructura while() se ejecuta dentro de la función setup(). La condición es que solo se dibuje dentro del lienzo.

Utilice variables que puedan ayudar a la construcción del dibujo, por ej: x, y, anchoEscalon, altoEscalon, etc.

El lienzo debería quedar como se muestra en la figura:



**Ejercicio 5:** Utilizando la estructura de control repetitiva **do-while**. Replique la siguiente imagen



La imagen debe ser construida desde la función `setup()`.

Defina el tamaño del lienzo en `size(600,600)`, verticalmente se divide el lienzo en franjas de igual medida, se deben dibujar los círculos sobre cada línea de por medio es decir en la línea 1 se dibujan círculos con distanciamiento, en la línea 2 no se dibuja y así sucesivamente. Las líneas tienen un color fijo, los círculos asumen colores aleatorios.