

# **FUNDAMENTOS DE LA PROGRAMACION ORIENTADO A OBJETOS**

**Trabajo práctico/Actividad**

**N°2**

**asistencia**

**Apellido y Nombre – LU/**

Calatayud, Alex Gabriel

TUV000576

**Grupo:**

**Integrantes**

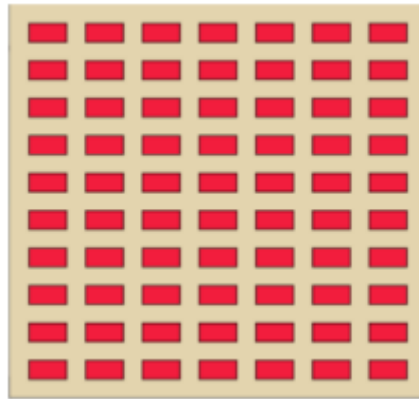
**AyN /LU**

**Profesor:**

Mg. Ing. Ariel Alejandro Vega

**Año: 2024**

**Ejercicio 20:** 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:



#### Análisis:

**Datos de Entrada:** Rectángulos dibujados en el lienzo según las especificaciones dadas.

**Datos de Salida:** Rectángulos dibujados en el lienzo según las especificaciones dadas.

#### Proceso:

**¿Quién debe realizar el proceso?:** El proceso puede ser realizado por un programa como processing.

**¿Cuál es el proceso que resuelve?:** dibujar una serie de rectángulos en un lienzo de tamaño específico, manteniendo una distancia específica entre ellos tanto horizontal como verticalmente, definiendo un bucle **for** para dibujar los rectángulos en el lienzo.

#### Diseño:

**Entidad que resuelve el problema:** lienzo

#### Variables:

- **coordenadasRect:** float //almacena un valor de coordenadas
- **ancho, alto, distanciaEntreRect :** int //almacena un valor entero
- **anchoLienzo, altoLienzo:** int //almacenan valores enteros

**Nombre del Algoritmo:** rectangulos\_repetidos

### Proceso del algoritmo:

1. inicio
2.  $\text{anchoLienzo} \leftarrow 440$
3.  $\text{altoLienzo} \leftarrow 420$
4.  $\text{ancho} \leftarrow 40$
5.  $\text{alto} \leftarrow 20$
6.  $\text{distanciaRect} \leftarrow 20$
7. **para**  $x \leftarrow \text{coordenadasRect.x}$  **hasta**  $\text{anchoLienzo}$  **con paso**  $(\text{ancho} + \text{distanciaEntreRect})$
8. **hacer**
9. **para**  $y \leftarrow \text{coordenadasRect.y}$  **hasta**  $\text{altoLienzo}$  **con paso**  $(\text{alto} + \text{distanciaEntreRect})$
10. **hacer**
11. dibujar rectangulo en  $(x,y,\text{ancho},\text{alto})$
12. **fin\_para**
13. **fin\_para**
14. fin

### Captura de la codificación en lenguaje processing

```
1 int distanciaEntreRect;
2 int alto;
3 int ancho;
4 PVector coordenadasRect;
5
6 void setup() {
7   size(440, 420);
8   background(235,235,200);
9   distanciaEntreRect = 20;
10  ancho = 40;
11  alto = 20;
12  coordenadasRect = new PVector(distanciaEntreRect, distanciaEntreRect);
13 }
14
15 void draw() {
16   dibujarRectangulo();
17 }
18
19 void dibujarRectangulo() {
20   for (float y = coordenadasRect.y; y < height; y += (alto + distanciaEntreRect)) {
21     for (float x = coordenadasRect.x; x < width; x += (ancho + distanciaEntreRect)) {
22       rect(x, y, ancho, alto);
23       fill(255,0,0);
24     }
25   }
26 }
```



