

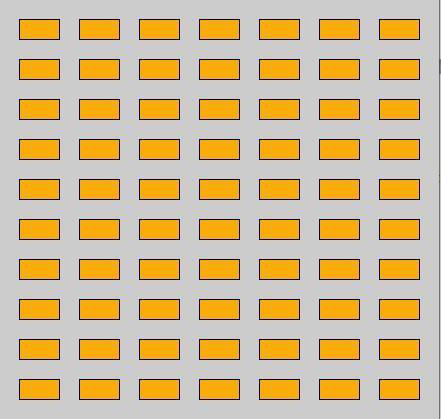
Ejercicio De Asistencia 09/04

Olarte Tatiana Magdalena

TUV00067

**EJERCICIO:**

* **Ejercicio 3:** 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 utilizando estructuras iterativas para que se vea de la siguiente manera:



**DESARROLLO:**

* **Ejercicio 3:**

**Definición del Problema**: Se debe dibujar en todo el lienzo rectángulos que tengan la misma medida, así como también una distancia de 20 pixeles tanto vertical como horizontal.

**Análisis:**

* Datos de Entrada: El tamaño del lienzo (440, 420), las medidas de cada rectángulo (40 de ancho y 20 de alto) y la distancia ente cada uno (20 píxeles tanto horizontal como vertical).
* Datos de Salida: Todos los rectángulos que se pueden dibujar en el lienzo.
* Proceso:

¿Quién debe realizar el proceso?: El lienzo

* ¿Cuál es el proceso que realiza…? Se realiza con la estructura de control repetitiva **for** y con una estructura de control selectiva que es el **if,** para poder dibujar todos los rectángulos que se puedan en el lienzo.

**Diseño:**

|  |
| --- |
| **ENTIDAD QUE RESUELVE EL PROBLEMA**: Lienzo |
| **VARIABLES**  altoRect: Real // almacena el alto de los rectángulos  anchoRect: Real // almacena el ancho de los rectángulos  distanciaEnX: Real // almacena la distancia entre cada rectángulo en x  distanciaEnY: Real //almacena la distancia entre cada rectángulo en y  altoLienzo: Real // almacena la altura del lienzo  anchoLienzo: Real // almacena la anchura del lienzo  contRect: Real // almacena los rectángulos que se va haciendo  totalRect: Real // almacena el total de rectángulos que hay |
| **NOMBRE ALGORITMO**: dibujar\_rectangulos    **PROCESO DEL ALGORITMO**   * anchoLienzo ← 440 * altoLienzo ← 420 * anchoRect ← 40 * altoRect ← 20 * distanciaEnX ← 20 * distanciaEnY ← 20 * totalRect ← 70 * **para** contRect **←** 0 **hasta** totalRect **con paso** (incremento 1) **hacer** * Dibujar un rectangulo en (distanciaEnX,distanciaEnY) con dimensiones en ancho y alto * distanciaEnY ← distanciaEnY + altoRect \* 2 * **si** (distanciaEnY > altoLienzo) **entonces** * distanciaEnY ← 20 * distanciaEnX ← distanciaEnX + anchoRect \* 1.5 * **fin\_si** * **fin\_para** |

**CONCLUSIÓN:**

Este trabajo me hizo saber más sobre cómo hacer un diseño sobre las estructuras repetitivas y selectivas, así como también poder programarlo.

**FUENTES BIBLIOGRÁFICAS:**

* clase 03 09 04 24 Fudnamentos de Programación Orientada a objetos: <https://youtu.be/YHpW_HMP_PE?si=PmhigKipfKlWiWaD>
* 14 Estructuras de Control Iterativas: <https://virtual.unju.edu.ar/mod/resource/view.php?id=300666>
* 11 Bifurcaciones - Expresiones lógicas - Estructuras selectivas: <https://virtual.unju.edu.ar/mod/resource/view.php?id=299585>