|  |  |  |
| --- | --- | --- |
| Texto  Descripción generada automáticamente con confianza media | TECNICATURA UNIVERSITARIA EN DISEÑO INTEGRAL DE VIDEOJUEGOS  FACULTAD DE INGENIERÍA  Universidad Nacional de Jujuy |  |

*Profesor:*

*Mg. Ing. Ariel Alejandro Vega*

*Año 2024*

Trabajo Práctico

N° 1

Apellido y Nombre – LU /

Aramayo Matias Nahuel

TUV000491

**FUNDAMENTOS DE PROGRAMACIÓN ORIENTADA A OBJETOS**

Punto 19: 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

Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente

Desarrollo del punto

**Análisis:**

**Datos de entrada:**

posY, incremento, distanciaRadio: int

**Datos de salida:**

Dibujar un circulo que ascienda y descienda

**DISEÑO**

|  |
| --- |
| Entidad: Lienzo |
| Variable:   * posY, incremento int * distanciaRadio=40 |
| Nombre del algoritmo: dibujar\_un circulo\_que\_ascienda\_y\_descienda   1. Leer posY=0 2. Leer incremento=1 3. Leer distanciaRadio=40 4. Dibujar Linea(0,posY,width,posY) 5. Dibujar circulo (width/2,posY+distanciaRadio,80,80) 6. posY+=incremento 7. si (posY>=height || posY<=0) 8. incremento\*=-1; 9. distanciaRadio\*=-1; 10. fin |

Fuentes bibliográficas: Archivos y PDF del profesor e ing Vega Ariel