



Lógica de programación

Fecha: sábado 14 de junio del 2025.

Título de trabajo:

Diagramas de flujo

Atari Pong es considerado uno de los juegos más influyentes en la historia de los videojuegos. Fue lanzado en 1972 y se convirtió en un éxito instantáneo. A continuación, se presentan algunos puntos clave sobre el desarrollo de Atari Pong:

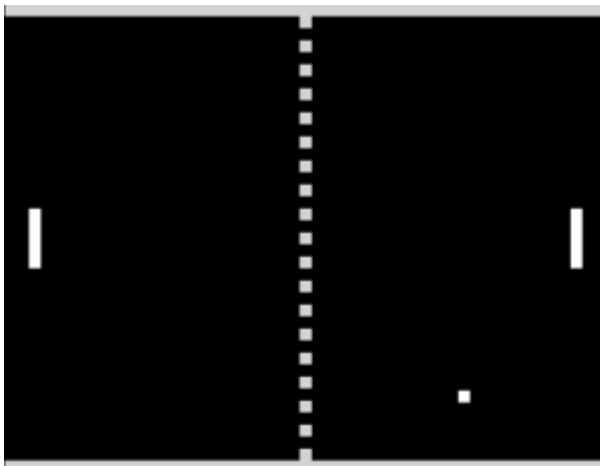
El siguiente hito importante en la historia de Atari es su entrada en el mercado doméstico. En 1973, Harold Lee sugirió la creación de una versión doméstica de PONG. Atari comenzó el diseño del proyecto Darlene en 1974. Harold Lee, Alan Alcorn y Bob Brown diseñaron el prototipo. Dado que Magnavox Odyssey ya se estaba quedando obsoleto y, de alguna manera, los clientes lo habían dejado de lado, los vendedores no se sintieron atraídos por el nuevo juego PONG de Atari hasta que Sears lo vendió bajo su sello Tele-Games en la Navidad de 1975. Tras este gran éxito, Atari lanzó su propia versión de PONG en 1976. El sistema contaba con una característica importante que la mayoría de los demás no tenían en 1975: el uso de un único chip que proporcionaba a los juegos puntuación digital en pantalla y un sonido atractivo. De hecho, otros sistemas aún utilizaban circuitos analógicos o digitales con componentes discretos. La puntuación digital en pantalla habría requerido más componentes en los circuitos, lo que habría supuesto un aumento del precio de venta. Gracias al diseño de un chip PONG especial por parte de Atari, el sistema pudo venderse a un precio normal con funciones avanzadas.

Con información requerida y obtenida a través de sitios web he podido obtener mucho ayuda para poder mejorar en el desarrollo del software para que este se puede desarrollar de una muy buena forma.

El desarrollo de Atari Pong fue un proceso rápido y sencillo que se basó en la creatividad y la innovación de sus creadores. La simplicidad y la adicción del juego contribuyeron a su éxito instantáneo y a su influencia en la industria de los videojuegos.

Juego básico de Pong en HTML y JavaScript

Esta es una implementación básica del juego Atari Pong, pero le faltan algunas cosas intencionalmente y se dejan como una exploración más profunda para el lector.



Exploración adicional

Puntaje

Cuando una pelota pasa por una paleta, el otro jugador debe anotar un punto. Usa `context.fillText()` para mostrar la puntuación en la pantalla.

Soporte para móviles y pantallas táctiles

Permite reducir el tamaño del juego al de un teléfono.

Consulta <https://codepen.io/straker/pen/VazMaL>

Admite controles táctiles

Trayectoria de la pelota

La bola debe cambiar de trayectoria según el punto donde impacta la pala. Por ejemplo, si impacta en la parte superior de la pala, debe tener un ángulo agudo hacia arriba, mientras que si impacta en el centro, debe moverse completamente plana hacia el otro jugador.

Nota importante: Responderé preguntas sobre el código, pero no añadiré más funciones ni responderé preguntas sobre cómo añadir más funciones. Esta serie pretende ofrecer una descripción general del juego, pero nada más.

Con ayuda de la información del anterior parte fue la guía para así mejorar el desarrollo de este y comprender de mejor manera lo que estoy realizando con la ayuda de esta guía he logrado obtener así un mejor esquema de la estructura que realizare esto.

En esta guía, aprenderá a programar una versión básica de Pong en Java. En esta versión, el juego se jugará contra el ordenador, en lugar de contra otro jugador. El juego se desarrollará con prácticas orientadas a objetos, por lo que podrá ampliarse o modificarse con relativa facilidad en el futuro.

import turtle

import time

Create screen

```
sc = turtle.Screen()
```

```
sc.title("Pong game")
```

```
sc.bgcolor("white")
```

```
sc.setup(width=1000, height=600)
```

Left paddle

```
left_pad = turtle.Turtle()
```

```
left_pad.speed(0)
```

```
left_pad.shape("square")
```

```
left_pad.color("black")
```

```
left_pad.shapesize(stretch_wid=6, stretch_len=2)
```

```
left_pad.penup()
```

```
left_pad.goto(-400, 0)
```

```
# Right paddle
```

```
right_pad = turtle.Turtle()
```

```
right_pad.speed(0)
```

```
right_pad.shape("square")
```

```
right_pad.color("black")
```

```
right_pad.shapesize(stretch_wid=6, stretch_len=2)
```

```
right_pad.penup()
```

```
right_pad.goto(400, 0)
```

```
# Ball of circle shape
```

```
hit_ball = turtle.Turtle()
```

```
hit_ball.speed(4) # Adjusted speed
```

```
hit_ball.shape("circle")
```

```
hit_ball.color("blue")
```

```
hit_ball.penup()
```

```
hit_ball.goto(0, 0)
```

```
hit_ball.dx = 5
```

```
hit_ball.dy = -5
```

```
# Initialize the score
```

```
left_player = 0
```

```
right_player = 0
```

```
# Displays the score
```

```
sketch = turtle.Turtle()
```

```
sketch.speed(0)
```

```
sketch.color("blue")
```

```
sketch.penup()
```

```
sketch.hideturtle()
```

```
sketch.goto(0, 260)
sketch.write("Left_player : 0   Right_player: 0",
             align="center", font=("Courier", 24, "normal"))
```

Functions to move paddles

```
def paddleup():
    y = left_pad.ycor()
    if y < 250: # Limit paddle movement
        y += 20
        left_pad.sety(y)

def paddleadown():
    y = left_pad.ycor()
    if y > -240: # Limit paddle movement
        y -= 20
        left_pad.sety(y)

def paddlebup():
    y = right_pad.ycor()
    if y < 250: # Limit paddle movement
        y += 20
        right_pad.sety(y)

def paddlebdown():
    y = right_pad.ycor()
    if y > -240: # Limit paddle movement
```

```
y -= 20  
right_pad.sety(y)
```

```
# Keyboard bindings
```

```
sc.listen()  
sc.onkeypress(paddleup, "w") # Changed to 'w'  
sc.onkeypress(paddleadown, "s") # Changed to 's'  
sc.onkeypress(paddlebup, "Up")  
sc.onkeypress(paddlebdown, "Down")
```

```
# Main game loop
```

```
while True:
```

```
    sc.update()  
    time.sleep(0.01) # Add delay to make game smoother
```

```
    hit_ball.setx(hit_ball.xcor() + hit_ball.dx)  
    hit_ball.sety(hit_ball.ycor() + hit_ball.dy)
```

```
# Checking borders
```

```
if hit_ball.ycor() > 280:  
    hit_ball.sety(280)  
    hit_ball.dy *= -1
```

```
if hit_ball.ycor() < -280:  
    hit_ball.sety(-280)  
    hit_ball.dy *= -1
```

```
if hit_ball.xcor() > 500:  
    hit_ball.goto(0, 0)  
    hit_ball.dy *= -1
```

```
    left_player += 1
    sketch.clear()
    sketch.write("Left_player : {}   Right_player: {}".format(
        left_player, right_player), align="center",
        font=("Courier", 24, "normal"))

if hit_ball.xcor() < -500:
    hit_ball.goto(0, 0)
    hit_ball.dy *= -1
    right_player += 1
    sketch.clear()
    sketch.write("Left_player : {}   Right_player: {}".format(
        left_player, right_player), align="center",
        font=("Courier", 24, "normal"))

# Paddle ball collision
if (hit_ball.xcor() > 360 and hit_ball.xcor() < 370) and \
    (hit_ball.ycor() < right_pad.ycor() + 50 and hit_ball.ycor() > right_pad.ycor() - 50):
    hit_ball.setx(360)
    hit_ball.dx *= -1

if (hit_ball.xcor() < -360 and hit_ball.xcor() > -370) and \
    (hit_ball.ycor() < left_pad.ycor() + 50 and hit_ball.ycor() > left_pad.ycor() - 50):
    hit_ball.setx(-360)
    hit_ball.dx *= -1
```

este es un código con el cual me estoy basando para tener una guía para realizar la estructura es una buena fuente de ayuda pero estoy tratando de implementar lo aprendido en la clases para que sea más fácil de entender.

Bibliografía:

<https://www.geeksforgeeks.org/python/create-pong-game-using-python-turtle/>

<https://karpathy.github.io/2016/05/31/rl/>

<https://gist.github.com/straker/81b59eecf70da93af396f963596dfdc5>

https://atari.fandom.com/wiki/Atari_Pong

