



# MULTI-DIMENSIONAL PONG

AUTHOR: CAMERON CUZMANKO

# HISTORY OF PONG

- Invented by Allan Alcorn in 1972
- Pong was initially meant to be a programming “warm-up”
- Began solely as an arcade game
- In 1974, engineer Harold Lee proposed a home Pong
- Sears signed an exclusive deal to sell Atari’s new Pong game
- Out by 1975 holiday season



# MOMENTUM

- Mass in motion
- Any object that is in motion has momentum
- $\vec{p} = m\vec{v}$ 
  - $p$  is momentum,  $m$  is mass, and  $v$  is velocity
  - Momentum can be thought of as a vector
- Impulse-Momentum Theorem
  - $\vec{F}\Delta t = m\Delta\vec{v}$  or  $\vec{F}\Delta t = \Delta\vec{p}$

# CONSERVATION OF MOMENTUM

- In a collision, the force exerted on object 1 is equal and opposite to the force exerted on object 2
  - Also true for the times of collisions (thus the impulses,  $\vec{F}\Delta t$ , are equal)
- Applies even when there is energy loss
- Fundamental to the physics of collisions

# TYPES OF COLLISIONS

- Inelastic Collisions
  - Momentum is conserved, BUT not all kinetic energy is conserved and some kinetic energy is changed into other forms of energy
    - i.e. Heat, sound, etc.
- Perfectly Inelastic Collisions
  - Occur when the lost kinetic energy is at a maximum

# TYPES OF COLLISIONS

- Elastic Collisions

- Momentum is conserved, some kinetic energy is still lost but the loss is negligible or nonexistent
- Conservation of momentum:  $m_1v_1 + m_2v_2 = m_1v'_1 + m_2v'_2$
- Conservation of kinetic energy:  $\frac{1}{2}m_1v_1^2 + \frac{1}{2}m_2v_2^2 = \frac{1}{2}m_1v'^2_1 + \frac{1}{2}m_2v'^2_2$

# REPRESENTING COLLISIONS IN CODE

- Velocity can be represented as a vector in code
- Treat collisions with the wall as rebound collisions
- Set conditions that tell the computer when a collision is required
- Depending on where the ball hits, you may simply switch the sign of a velocity component
- This represents a theoretical conservation of momentum and energy

# PONG AI (ARTIFICIAL INTELLIGENCE)

- Artificial Intelligence is used to generate behaviors in non-player characters that exhibit human-like intelligence
- Created so that players could play games, like pong, without a second player and still feel like others are around
- Game AI is very different from standard Artificial Intelligence



# PONG AI (ARTIFICIAL INTELLIGENCE)

- Need to implement an intentionally imperfect AI
  - If the AI can hit the ball back every time or not at all, the game is no fun
    - Need a happy medium
- Best option is to have the paddle track the ball with a set maximum speed

# PONG AI (ARTIFICIAL INTELLIGENCE)

- Need to implement changing ball speed to beat AI
  - X and Y velocities increase at a constant multiple depending on where the collision occurs
    - Edge of the paddles increase the velocity while the centers do not
  - Z increases with every paddle hit

# USER CONTROL

- Game requires user interaction in order to be worth playing
- Used VPython's `getkey()` function to read user keypresses
  - Made a method that moved the paddle based on the user's keypress
- Controls:
  - W=up, S=down, A=left, D=right, Q=into the screen, E=out of the screen, and escape=end game

# FURTHER IMPROVEMENTS

- Playable 4D (3D representation of 4D) stage
- More randomization in the initial velocity of the ball
- Better AI that allows for more variety
- Rotatable user paddle