ThanosDiesInEndgame

Team 7

PONG 2.0

LIAD H, RITHVIK M, EYAL S

Project Overview

Our program is a recreated and re-imagined version of the classic game; Atari Pong. It will provide the average person who is bored with a leisurely (or staggeringly difficult) gaming experience. Anybody looking for a quick and casual game can turn to Pong 2.0.

Program Overview

Game Controls:

- Press W to move up, and S to move down
- Press P to pause and spacebar to enter the shop
- Don't let the ball touch your wall!
- The red squares in hard mode kill you when you touch them

Bugs:
When the ball gets wedge between the paddle and the wall or close to top/bottom of the paddle, the ball gets stuck there until all of your health is depleted.

Code Overview

- Game: represents a game, in this case Pong 2.0
- GameObject: responsible for creating the game objects like the player, ball, etc.
- Handler: represents a handler for private variable actions
- Window: represents a JFrame window display
- MouseManager: responsible for handling mouse events
- Player: represents the paddle
- Ball: represents the ball
- Enemy: represents the red enemies in hard mode

Code Overview Contd.

- Menu: represents the main menu screen
- MenuParticle: represents the animated particles in the main screen
- Spawn: spawns enemies
- Shop: represents the game shop to upgrade/refill health and upgrade speed
- Trail: represents the menu particle’s trail
- ID: holds the data for various objects
- HUD: represents the health bar, pause, and shop options
- KeyInput: handles the key pressing logic if multiple keys are pressed at the same time

Structure

- Object classes (Ball, Player, etc) extend an abstract class; GameObject.
- Abstract Methods:
  - tick()
  - render(Graphics g)
  - getBounds()

Static Fields

Game:

- serialVersionUID (also in Window)
- WIDTH and HEIGHT
- paused
- gameState

HUD:

- HEALTH

Menu:

- clip(also in Ball)

Challenges

One of the hardest challenge that we encountered was syncing our workspaces. Initially we tried using GitHub, but it wasn’t good when using school computers because some features were blocked. We solved this problem by creating a folder in google drive and then creating subfolders for each of us and sync from there.

Next Steps

If we had more time, we would:

- Add a second player
- Create different levels
- A CPU player option

Demo

Q and A's

```
System.out.println("Thank You");
```