GE Final Practical Explanation

Oscar Gao 100866004

Command Pattern
- Player Commands
    - A command was created for each player action for movement.
      MoveLeftCommand, MoveRightCommand, and JumpCommand. This was so I
      could encapsulate all movement logic into one script. I could also handle the
      inverted movement logic in these scripts. First I created an interface for
      PlayerCommand and then made several concrete commands for movement. In
      these commands I handled the movement logic and handled the inverted
      movement logic. In the PlayerController script, I invoked the Execute() method for
      each command when I needed to use them.

Object Pooling and Singleton
- Snowflake projectile pool
    - I made a ProjectilePool to handle the projectiles created by snowflakes. I made a
      ProjectilePool class with a GetFromPool() and ReturnToPool() method. I also
      made this object pool class a singleton so I could access it anywhere. I did this
      so I could use the pool without needing to reference it in the Snowflake and
      Projectile scripts. The ProjectilePool script utilizes a Queue and returns and takes
      from the queue as projectiles are needed. As they are returned and taken from
      the queue, they are activated and deactivated respectively.

Diagram Below

Prime: Object Pooling

Projectile

Singleton<T>

+ Instance

ProjectilePool

Queue<T> pool

+ GetFromPool()
+ ReturnToPool(obj)

Snowflake