CSE 2100 Polymorphism Articulate

Zach Newby June 28, 2023

Polymorphism is the ability to overwrite or modify inherited methods or attributes in classes, allowing for increased functionality and versatility, without having to code an entire class from scratch. There are several ways to use polymorphism, you could use virtual statements in the base class, and override or new statements in the child class, or even use an abstract class or interfaces for the base class.

Polymorphism’s main benefit is creating subclasses that have different functions and uses from the base class while still adhering to the Liskov Substitution Principle. For example, a video game with multiple playable characters, where certain characters would overwrite the methods from the base class, allowing them to have different interactions with certain objects without breaking the game.

An example of my use of polymorphism is in my Eternal Quest Program, where each goal subclass could be saved and loaded, but due to a couple subclasses having different attributes from the base class I had to overwrite the save and load methods to account for them.

Goal Base Class GetSaveData method:

virtual public String GetSaveData(){

        return $"{\_type}|{\_name}|{\_rewardPoints}|{\_description}|{\_status}";

    }

ChecklistGoal Child class GetSaveData method:

override public String GetSaveData(){

        return $"{\_type}|{\_name}|{\_rewardPoints}|{\_description}|{\_status}|{\_timesCompleted}|{\_timesToComplete}|{\_bonusRewardValue}";

    }