Assignment 3

Mike Peters
petermi3@oregonstate.edu
11/08/2015

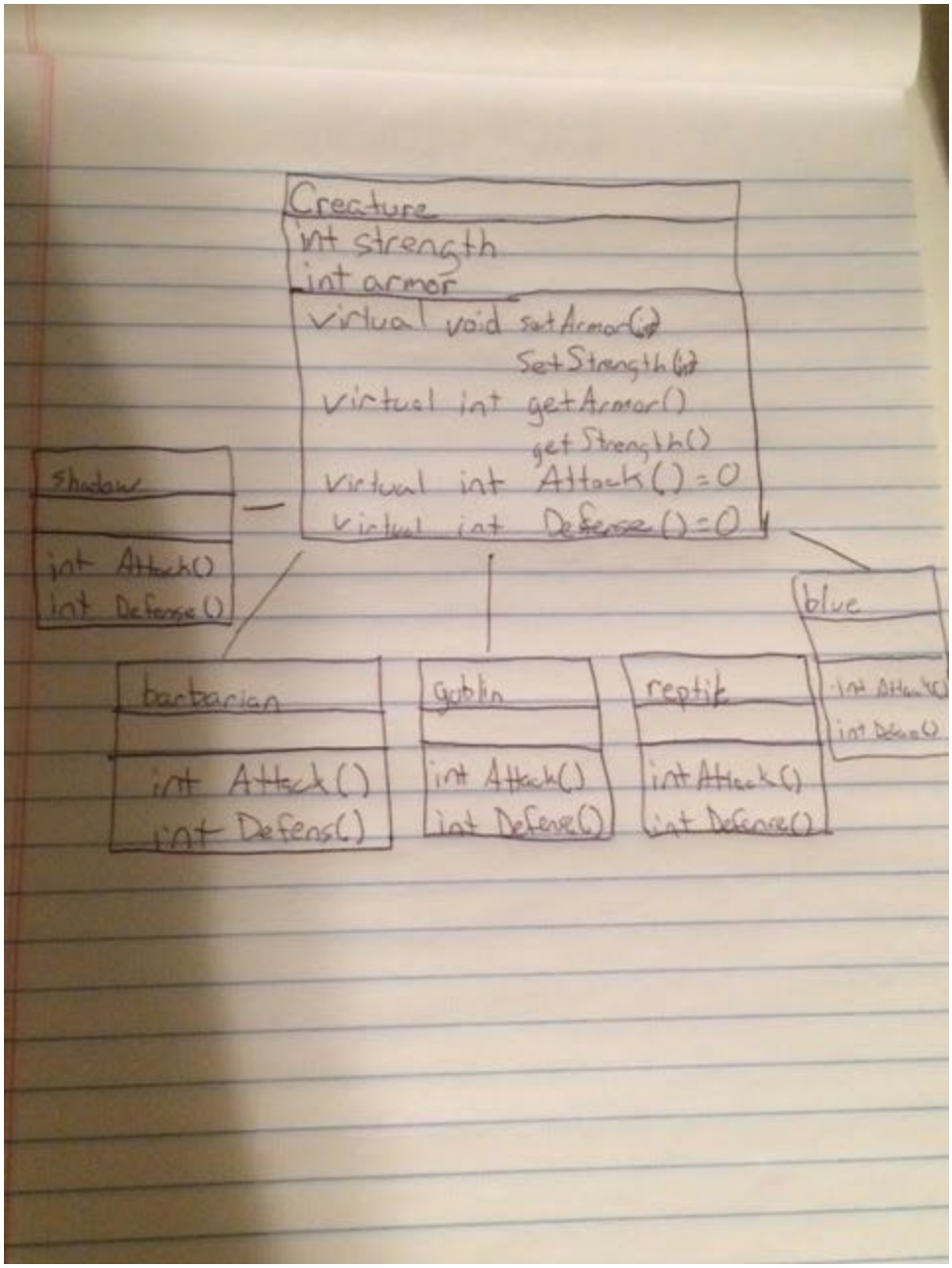
Design

You will create a simple class hierarchy as the basis for a fantasy combat game. Your 'universe' contains **Goblins**, **Barbarians**, **Reptile People**, **Blue Men** and **others**. Each will have **characteristics** for **attack**, **defense**,**armor**, and **strength** points.

To resolve an **attack** you will need to generate 2 dice **rolls**. The attacker rolls the appropriate number and type of dice under **Attack**. The defender rolls the appropriate number and type of dice under **Defense**. **You subtract the Defense roll from the Attack roll. That is the damage. To apply the damage you subtract the Armor value. The result is then subtracted from the Strength Points.** That value becomes the **new Strength Points** for the next round. If **Strength** Points goes to **0 or less** then the character is out of the combat. Remember that for **The Shadow** there may be 0 points inflicted based on his **special ability**. You need to create a **Creature** class. Then you will have a subclass for each of these characters. Note that the Creature class will be an **abstract class**. You will never instantiate one. For our purposes right now each subclass will vary only in the values in the table. Since each starts with the same data elements you will only need one constructor. It is part of your design task to determine what functions you will need. The only value that can change is the **Strength** Points.

Test Plan

Test Case	Input Values	Expected Outcomes
barbarian vs barbarian	fix dice so player one wins on first throw.	output of "Player 1 wins"
barbarian vs barbarian	fix dice so player one wins on first throw.	output of "Player 2 wins"
goblin vs reptile	random	display each round of attacks until 0 strength is reached. strength is displayed after each attack.
blue men vs the shadow	makes special ability enabled 100% of the time	the shadow will eventually always win. displayed damage done by blue men will always be 0
display armor and strength of each creature type when created.	create a creature of each type then cout the armor and strength values	these number should match the table provided in the assignment


Creature
int strength
int armor
virtual void setArmor()
SetStrength()
virtual int getArmor()
getStrength()
virtual int Attacks() = 0
virtual int Defense() = 0
Shadow
int Attack()
int Defense()
blue
barbarian
int Attack()
int Defens()
Goblin
int Attack()
int Defense()
reptik
int Attack()
int Defense()

Taking the time to plan this out and break it down made it seem really easy. Once you got one class working the rest were just a matter of changing the data to match each creature in the provided table. Putting all the class declarations in a single header file made sense to me, so I did to just to prevent having so many files. Testing each creature after I implemented (simple displays of each value after it was created) it proved useful and caught minor logic or syntax errors. Within the game a made each attack display how much damage was dealt, and what the new strength would be. This helped check that the math logic was working as intended. I feel like I am getting more comfortable with pointers. This has been a fun assignment and I look forward to adding to it.