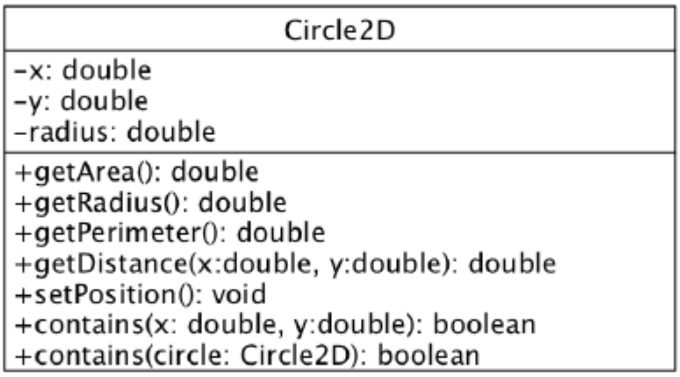
CS 249: Assignment 5

Thinking in Objects

Theory Questions (14%)

Note: for any UML diagrams, you can use a UML drawing utility like UMLet.

1. (2%) Draw the UML diagram for the class defined in Programming Exercise 10.11 at the end of Chapter 10. **Enforce encapsulaton!**



Circle2D

-x: double
-y: double
-radius: double

+getArea(): double
+getRadius(): double
+getPerimeter(): double
+getDistance(x:double, y:double): double
+setPosition(): void
+contains(x: double, y:double): boolean
+contains(circle: Circle2D): boolean

2. (2%) Design the following two classes for a top-down 2D game (similar to the NetHack example from before).

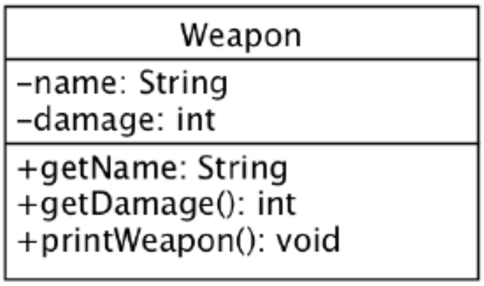
You will draw UML diagrams for both of these classes.

First, design a class for a **Weapon**.
It should contain the following data:

- Name
- Damage

It should have the following functionality:

- Constructor that takes name and damage
- Getter/setter functions for data
- STATIC function that takes a Weapon and prints the name (if null, print "No weapon")



Weapon

-name: String

-damage: int

+getName: String

+getDamage(): int

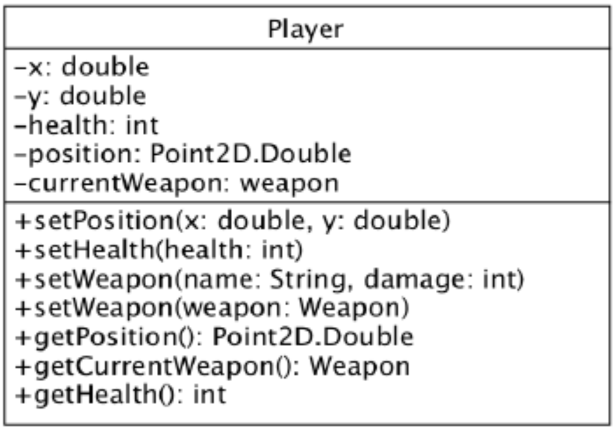
+printWeapon(): void

Next, design a class for a **Player**.
It should contain the following data:

- Position in X and Y (you may assume integer coordinates)
- Health (defaults to 100)
- Current Weapon (defaults to null)

It should have the following functionality:

- Constructor that takes position
- Getter/setter for position
- Getter/setter for health
- Getter/setter for current Weapon



Player

-x: double

-y: double

-health: int

-position: Point2D.Double

-currentWeapon: weapon

+setPosition(x: double, y: double)

+setHealth(health: int)

+setWeapon(name: String, damage: int)

+setWeapon(weapon: Weapon)

+getPosition(): Point2D.Double

+getCurrentWeapon(): Weapon

+getHealth(): int

Ensure that your classes are:

- Clear, consistent, and have central purpose
- Maintaining a good abstraction
- Enforcing encapsulation

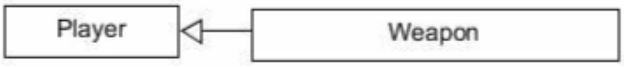
3. (2%) Given the Player and Weapon classes (also look at Programming Requirement 2), what do you think most accurately represents the relationship between Player and Weapon?

   (a) Association (b) Aggregation

   (b) Aggregation

   (c) Composition

4. (2%) Given your answer, draw the UML diagram that shows the relationship between these classes. **Use a simplified form of the UML class diagram that just includes the name.** Also, **be sure to include the correct numbers on either end of the line!**



5. (2%) Which of the following is **NOT** part of a **class contract**?

   (a) Description of how public parts are expected to behave

   (b) Private methods and fields

   (c) Public constructors, methods, and fields

   (a) Description of how public parts are expectected to behave

6. (2%) An **Abstract Data Type (ADT)** is 1) a collection of data and 2) operations that work on that data.

   (a) True (a) True

   (b) False

7. (2%) Given the code below, what is this an example of?

   (a) Autoboxing (a) Autoboxing

   (b) Autounboxing

   Integer x = 5;