

CSCI 306 Final Exam

Fall 2019

100 Points

Name: Adam Stogsdill

BE SURE TO READ THE INSTRUCTIONS BEFORE YOU BEGIN! These questions will not make sense unless you have read that document, as it contains additional requirements.

General points for design, clarity, completeness, grammar, spelling, English, etc. – 10 pts

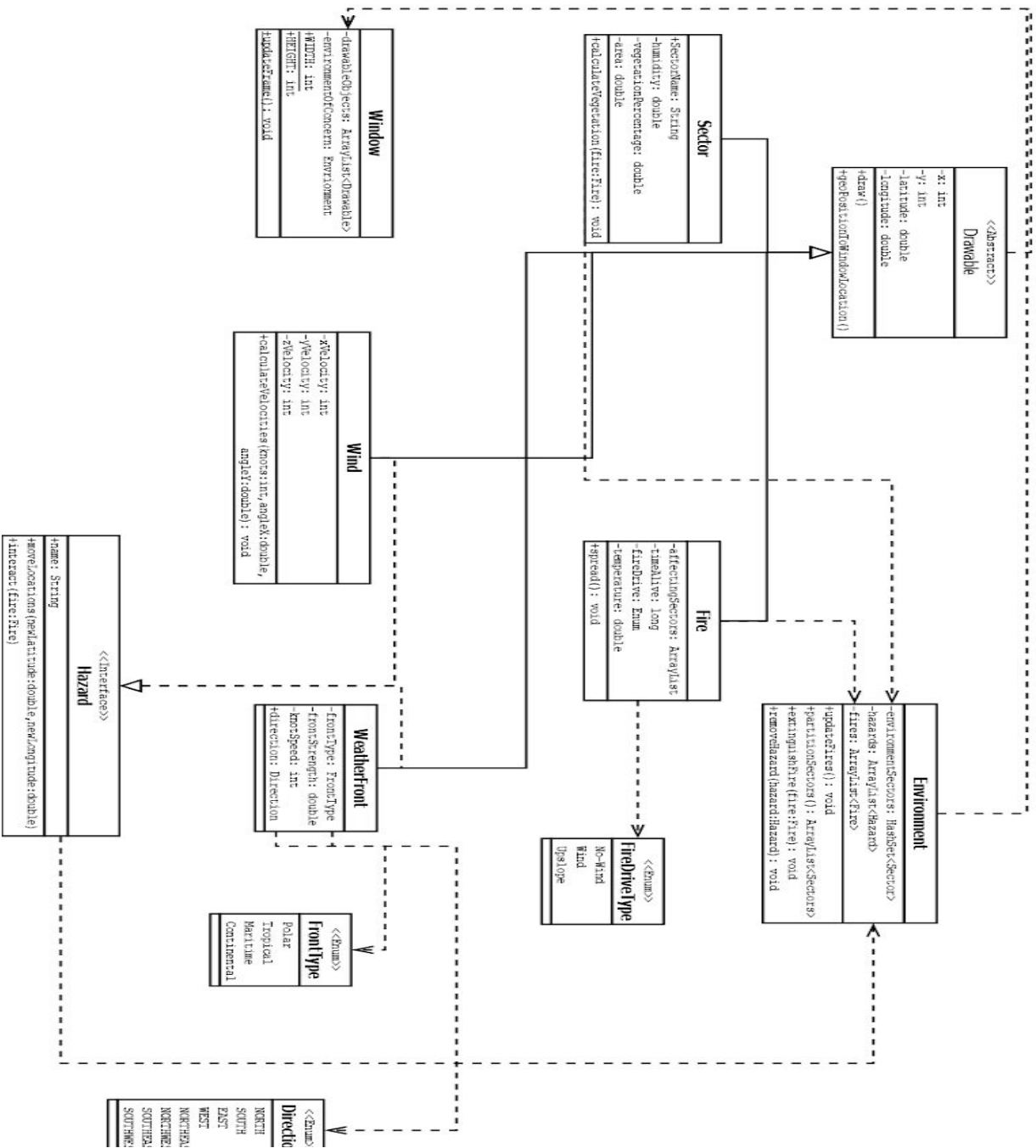
1. Provide a brief explanation of your program and the subsystem you are modeling. What are the main aspects, requirements and goals? (10pts)

My program is designed to provide a graphical interface and fire behavior model that separates the environment into sectors and calculated the fire's path and the importance of each sector to help people mitigate the fire and eventually stop its movement.

The program needs some graphical user interface otherwise the behavior model will not make much sense if printed through a console. It needs to be able to monitor the fire, it needs to be able to take into account other factors besides just the fire to show this behavior, and it needs to be able to be updated and constantly calculate this information in real-time.

2. Insert your UML diagram of your design below. Be sure that the diagram is readable! Break it up into a couple of pieces if needed. Because this is take home, links and format of UML need to be correct. The following answers need to be consistent with the UML design. (20pts)

Fire Behaviour Model



3. What are your abstract class(es)? Choose one and explain *why* your abstract class should be abstract, and why you need child classes. (10pts)

My abstract class is Drawable because every object that needs to be drawn on the screen from the geographical location however each of the child classes will have different ways of drawing themselves so this way I can implement a default method for the simpler objects but also override this method when I have objects like sector that need to draw a specific area.

4. What are your interface(s)? Choose one and justify *why* it makes sense to include that interface. Do NOT just grab a function from one of your classes and throw it into an interface. Your design and explanation should clearly show when an interface is appropriate compared to an abstract class. (10pts)

My Hazards interface because all of the objects should be able to be moved in a systematic way and also each of these classes need to provide interactions with other hazards. Each class will have different interactions thus I don't need an initial implementation of these variables or methods which is why I chose to use an interface instead of an abstract class.

5. Explain how your design adheres to the Open-Closed Principle (OCP). **Be specific.** (10pts)

My program adheres to OCP because all of the objects that implement or extend another class will allow the program to have extensions to the code through the objects without the original abstract or implementation having to be edited. For instance, the Drawable abstract class doesn't have to be adjusted itself if we want to add another class that can implement Drawable instead we just adjust the original methods to perform the same function in the child class but allow .

6. Explain how your design adheres to the Interface Segregation Principle (ISP). **Be specific** (e.g., you might describe a naïve version of your design that would violate ISP, to show that yours does not violate ISP). (10pts)

I could combine both the Hazard Interface and the Drawable object however Sectors don't move because sectors are calculated when the environment object is created. Thus sectors shouldn't implement calculateVelocities(). However any object that needs to be drawn needs a draw method and also needs a geoPositionToWindow().

7. Explain how someone might extend your program in a way that violates Liskov Substitution Principle (LSP). **Be specific** (i.e., give a concrete example). (10pts)

If the WeatherFront class implemented the geoPositionToWindowLocation but required a third parameter into the function (i.e., z or positionFromTheGround) then it would violate Liskov Substitution Principle because we would inherently be changing the function of

geoPositionToWindowLocation and would increase the restrictions given by the parameters and would have to behave differently from the original source function.

8. Include code for one failing JUnit test method. Include a brief comment about what you are testing. Remember that TDD is a design process. This method should test some planned functionality (e.g., test that a car slows down if another car's brake lights are showing), and NOT just a setter/getter. (10pts)

@Test

```
    public void testLatitudeLongitudeConversion() {  
        //I am trying to test if the conversion between the latitude and the longitude  
        reveals the expected x and y variables based off the object's parameters  
  
        //of our window size and the fire's latitude and longitude.  
  
        Window w = new Window(500, 500);  
  
        ArrayList<Sectors> affectingSectors = new ArrayList<Sectors>();  
  
        Fire f = new Fire(affectingSectors, 0, FireDriveType.NOWIND, 753.5,  
39.7605774, -105.2453558);  
  
        int desiredX = 200;  
  
        int desiredY = 400;  
  
        int[] geoPositiontoXY = f.geoPositiontoWindowLocation();  
  
        assertEquals(desiredX, geoPositionXY[0]);  
  
        assertEquals(desiredY, geoPositionXY[1]);  
  
  
        Fire f = new Fire(affectingSectors, 0, FireDriveType.NOWIND, 760.5,  
39.7605774, -105.2453558);  
  
        int desiredX = 210;  
  
        int desiredY = 402;  
  
        int[] geoPositiontoXY = f.geoPositiontoWindowLocation();  
  
        assertEquals(desiredX, geoPositionXY[0]);  
  
        assertEquals(desiredY, geoPositionXY[1]);  
  
    }
```

