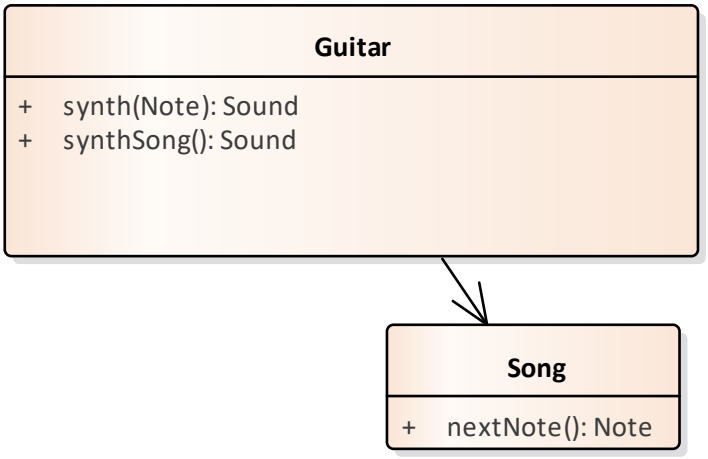
Half-Exam 1 Name:

No note-sheets, calculators, etc. on this exam. Please read the whole exam before you get started.

1. (30 points) Consider the diagram below. To synthesize a complete song, the guitar synthesizes each note from the song and concatenates the sounds together. ***Edit*** the diagram below so that the Guitar can change its behavior for synthesizing notes. Sometimes, it will synthesize each note using a simple Pluck and sometimes using a Strum (rubbing over the strings many times for a single note). In the future, other behaviors may be added like Pitch Bend (moving the fingers on the frets to slide the pitch during the note), but you do not need to include this behavior. Be sure to ***include*** a method that allows the synthesizing behavior to change. ***Edit*** the UML diagram **in detail** to illustrate the Strategy Pattern.



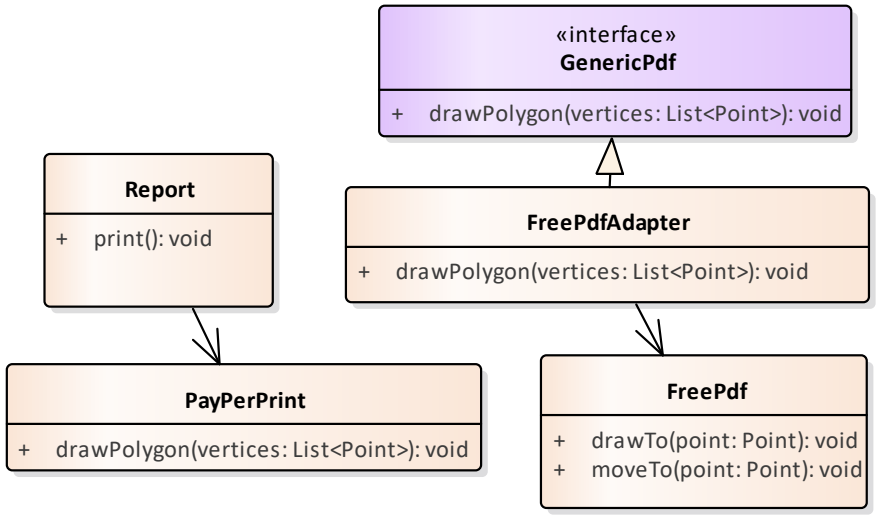
2. (10 points) Consider a Vehicle class with make, model, year, and VIN (Vehicle Identification Number) fields. ***Circle one***: this is a **Domain Object** / an **Attribute.** ***Explain*** your choice.

3. (10 points) Consider a GpsCoordinates class with latitude and longitude fields. ***Circle one***: this is a **Domain Object** / an **Attribute.** ***Explain*** your choice.

4. (10 points) ***Describe*** cohesion and coupling in your own words, making clear which is which.

5. (20 points) ***Give a code example*** illustrating coupling.

6. (20 points) Consider the partial implementation of the Adapter pattern illustrated on the right below. The adapter is meant to allow the FreePdf to replace the PayPerPrint pdf. However, the transition to the adapter is not complete. ***Edit*** the Report ***code*** shown on the left below to illustrate programming to an interface. Your solution should allow the Report to more easily be edited to use other adapters in the future.




```
public class Report {

    private PayPerPrint printer;

    public Report(PayPerPrint p) {
        this.printer = p;
    }

    public void print(){/*…*/}
}
```