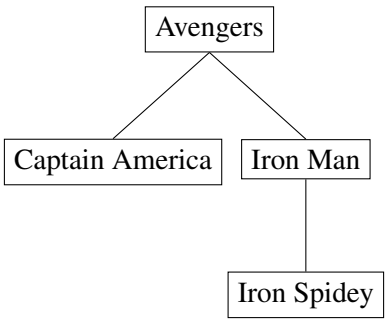
CS 61B
Spring 2019

Extra Practice Worksheet
Discussion 4: Inheritance

Worksheet 4 Solutions

1 Infinity War

Consider the hierarchy of classes shown in the following diagram.



Avengers is specified as an abstract class as shown in the following declaration. Each avenger has a name that is specified upon construction.

```
public abstract class Avenger {
    private String name;
    public Avenger(String name) {
        this.name = name;
    }
    public String getName() {
        return name;
    }
    public abstract String fight();
}
```

The subclass Iron Man has the partial class declaration below.

```
public class IronMan extends Avenger {
    public IronMan(String avengerName) {
        /* implementation not shown */
    }
    public String fight() {
        /* implementation not shown */
    }
}
```

a. Given the class hierarchy shown above, write a complete class declaration for the class ”Captain America”, including implementations for its constructor and method(s). The ”Captain America” method fight returns ”We do not trade lives” when it is invoked.

Solution:

```
public class CaptainAmerica extends Avenger {
    public CaptainAmerica(String avengerName) {
        super(avengerName); }
    public String fight() {
        return "We do not trade lives";
        }
}
```

b. Given the declaration of the IronMan class, if the string ”magic” is returned by the IronMan method fight, then the IronSpidey method fight returns a String containing ”magic” repeated two times.

Given the class hierarchy shown previously, write a complete class declaration for the class IronSpidey, including implementations of its constructors and method(s).

Solution:

```
public class IronSpidey extends IronMan {
    public IronSpidey(String avengerName) {
        super(avengerName);
    }
    public String fight() {
        return super.fight() + super.fight();
    }
}
```

c. Consider the following partial declaration of the class AvengerHouse.

```
public class AvengerHouse {
    private List<Avengers> avengersList;

    /** For every Avenger in the AvengerHouse,
    prints the avenger name followed by the result
    of a call to its fight method, one line per
    avenger. **/
    public void assemble() {
     /* to be implemented in part c */
    }
}
```

Write the AvengerHouse method assemble. In writing this method, you may use any of the methods defined for any of the classes specified for this problem.

Solution:

```
public void assemble() {
    for (Avenger a : avengersList) {
        System.out.println(a.getName() + a.fight());
    }
}
```