Homework 5

1. In the last homework assignment, you have implemented a **Cycle** class and its subclasses **Unicycle**, **Bicycle**, and **Tricycle**. Here you need to create a **Clown** class. The **Clown** class has a method **check()**, which takes a **Cycle**. The **check()** method will check if a given Cycle is a **Tricycle**, and if it is a **Tricycle**, print out "I refuse to ride it"; otherwise, print out "I'm happy to ride it". (1 point)

2. Write a method **print()** in a class named **Hierarchy**, which takes an object and ***recursively*** prints all the class names in that object's hierarchy. The first printed class should be the root of the hierarchy and so forth (1 point)

3. Create an **IterableFibonacci** class, which implements **Iterable<Integer>** interface. The following is a code skeleton you can start with. Write a range-based for loop to test your implementation. (1 point)

```
// print out the first n numbers in the Fibonacci sequence:
// 0, 1, 1, 2, ...
public class IterableFibonacci implements Iterable<Integer> {
  private class FibIterator implements Iterator<Integer> {
    public boolean hasNext() {
      // has output n numbers yet?
    }
    public Integer next() {
        // the next number in the sequence
    }
    public void remove() { // Not implemented
        throw new UnsupportedOperationException();
    }
  }
  private int n;
  // add private fields as needed
  public IterableFibonacci(int n) {
    // initialize n
  }
  public Iterator<Integer> iterator() {
     return new FibIterator();
  }
}
```

4. Create a generic, singly linked list class called **SList**, which, to keep things simple, **does not** implement the List interface. Each **Link** object in the list should contain a reference to the next element in the list, but not the previous one. (2 points)