CISC 3115
Final Exam

Name:_____

Question 1 (25 points)

Does each of the following programs compile and run? If no, explain the reason. If yes, give the output.

1.
```
class P1 {
    public static void main(String[] args){
        show("hello");
    }

    public void show(String s){
        System.out.println(s);
    }
}
```

2.
```
class SuperP2 {
    void m(){
        System.out.println("m of SuperP2");
    }
}

class P2 extends SuperP2 {
    void m(){
        System.out.println("m of P2");
    }

    public static void main(String[] args){
        SuperP2 e = new SuperP2();
        e.m();
    }
}
```

3.
```
import java.util.*;

class P3 {
    ArrayList<Integer> lst = new ArrayList<>();

    public void add(Integer obj){
        if (!lst.contains(obj))
            lst.add(obj);
    }

    public void show(){
        for (Integer i: lst)
            System.out.println(i);
    }

    public static void main(String[] args){
        P3 d = new P3();
        d.add(1); d.add(1);
        d.add(2); d.add(2);
        d.show();
    }
}
```

4.
```
class P4 {
    public static void main(String[] args){
        Integer[] a = {1,2,3};
        int s = 3;
        for (int e : a){
            s += e;
        }
        System.out.println(s);
    }
}
```

5.
```
public class P5 {
    static void pretty_print(String s){
        int n = s.length();
        if ( n <= 3) {
            System.out.print(s);
        } else {
            pretty_print(s.substring(0, n-3));
            System.out.print("," + s.substring(n-3));
        }
    }

    public static void main(String[] args){
        pretty_print("1234567890");
    }
}
```

Question 2 (20 points)

Implement a class named `MySortedArrayList`, which extends `java.util.ArrayList<Integer>` for storing a list of integers in *ascending* order without duplicates. The class contains the following methods:

3.1. `public boolean add(Integer elm)`: If `elm` occurs in the list, this method does nothing and returns `false`; otherwise, it inserts `elm` into the list such that the list remains sorted in ascending order after insertion, and returns `true`. For example, if this list is `[1,5]`, then the call `add(3)` on the list changes the list to `[1,3,5]`.

3.2. `public MySortedArrayList union(MySortedArrayList lst)`: This method returns the union of this list and the given list `lst`. The returned list must be sorted in ascending order. For example, if this list is `[1,3,5]` and `lst` is `[3,4,5]`, then the returned list is `[1,3,4,5]`.

Question 3 (20 points)

Assume the availability of the following class `Rational`. Write a class named `ComparableRational` that extends the `Rational` class and implements the `Comparable` interface. The `compareTo` method compares `this` object with a given `ComparableRational` object mathematically. For example, $1/3 < 1/2$.

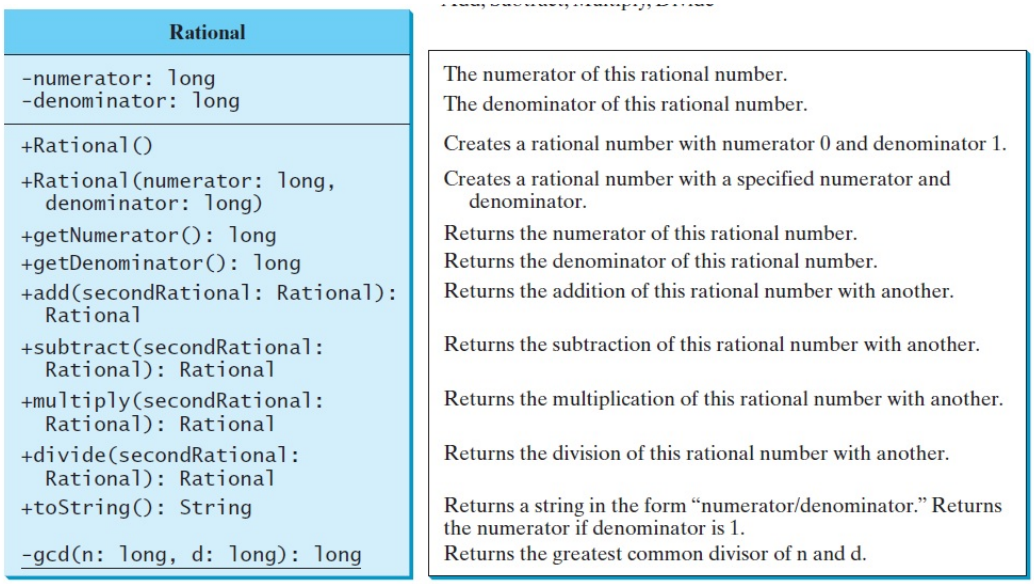

Figure 1: The `Rational` class

Question 4 (20 points)

In order to write a program to detect broken links, you need a function of the following specification:

```
public static ArrayList<String> hyperLinks(Scanner sc)
```

which takes a scanner that is linked to an HTML document, and returns a list of hyperlinks contained in the document. A hyperlink is defined by the HTML tag `<a>` with the following syntax:

```
<a href="url">link text</a>
```

For example, for the following document

```
<H3> Resources </H3>
<UL>
  <LI> <a href="http://docs.oracle.com/javase/tutorial/"> Java Online Tutorials</a>
  <LI> <a href="https://en.wikipedia.org/wiki/Object-oriented_programming"> OOP Wiki</a>
  <LI> <a href="https://www.courseduck.com/programming/java/"> CourseDuck's Java page</a>
</UL>
```

The returned list contains the following three hyperlinks:

```
"http://docs.oracle.com/javase/tutorial/",
"https://en.wikipedia.org/wiki/Object-oriented_programming"
"https://www.courseduck.com/programming/java/"
```

Implement the `hyperLinks` function.

Question 5 (15 points)

Consider the following program:

```
import java.math.BigInteger;

public class P {
    public static BigInteger f(BigInteger x){
        if (x.equals(BigInteger.ZERO))
            return BigInteger.ZERO;
        return f(x.divide(BigInteger.TEN)).add(x.mod(BigInteger.TEN));
    }

    public static void main(String[] args){
        BigInteger x = new BigInteger("123456789");
        System.out.println(f(x));
    }
}
```

1. What is the output of the program?
2. What does the function `f` return in general?
3. Rewrite the function `f` such that it uses iteration instead of recursion.

This page is blank.