Notes and Definitions

- **Abstraction Function:** Maps Concrete State to Abstract State
- **Representation Invariant:** Maps Concrete State to Boolean
  - TRUE iff Abstraction Function holds

Problems

1. Fill in the abstraction function and representation invariant for this implementation of Circle. Suppose our concrete representation in this case is two points directly across from each other, representing the endpoints of a diameter of the circle.

```java
public class Circle3 {
    private Point corner1, corner2;

    // Abstraction function:
    // AF(this) = a circle c with center (x,y) and
    // radius r such that
    //     (x,y) = _________________________________
    //     r = _________________________________

    // Rep invariant:
    // _________________________________
    // _________________________________
}
```

2. Given the following ADT, NonNullStringList, find two concrete representations for it. NonNullStringList is a list of string such that there are no null values in the list. Note your implementations must have
some way to implement the three abstract operations provided \((\text{add, remove, get})\). Write out the abstraction function and representation invariant for both.

**Hint**: Recall the two implementations of List.

**Concrete Representation 1:**
```java
public class NonNullStringList {
    // Abstraction function:

    // Rep invariant:

    // Fields:

    public void add(String s) { ... }
    public boolean remove(String s) { ... }
    public String get(int i) { ... }
}
```

**Concrete Representation 2:**
```java
public class NonNullStringList {
    // Abstraction function:

    // Rep invariant:

    // Fields:

    public void add(String s) { ... }
    public boolean remove(String s) { ... }
    public String get(int i) { ... }
}
```