Exercise Sheet 8

1. Are ownership type modifiers correctly used in the following program? Which expressions are legal? Which are invalid?

```
public class Assignments{

  rep      Integer attr_x;
  readonly Integer attr_y;
  peer     Integer attr_z;


}
```

```
public void m(readonly Integer a ) {
  rep      Integer x = new rep Integer(4);
  readonly Integer y = new     Integer(3);
  peer     Integer z = new     Integer(2);

           x = y;    x = z;    attr_x = x;
           y = x;    y = z;    attr_y = a;
}
```

2. Annotate the following program with the ownership type modifiers:

```
class Human {}
```

```
class Room {
  private Human m;
  void enter( Human m ) {/*...*/}
  Human leave() {/*...*/}
}
```

```
public class Building {
  private Room[] rooms;
  void enter(Human m) {rooms[0].enter(m);}
  Human leave() {return rooms[0].leave();}
}
```

3. Annotate the following program with the ownership type modifiers to maximize the buffer, the producer, and the consumer encapsulation:

```
class Producer {
  int[] buf;
  int n;
  Consumer con;
  Producer() {
    buf = new int[10];
  }
  void produce(int x) {
    buf[n] = x;
    n = (n+1) % buf.length;
  }
}
```

```
class Consumer {
  int[] buf;
  int n;
  Producer pro;
  Consumer(Producer p) {
    buf = p.buf;
    pro = p;
    p.con = this;
  }
  int consume() {
    n = (n+1) % buf.length;
    return buf[n];
  }
}
```

```
class Context {
  Producer p;
  Consumer c;

  Context(){
    p = new Producer();
    c = new Consumer(p);
  }

  public void run() {
    for(int i=-5; i <=5; ++i){
      p.produce(i);
      if(i%2 == 0)
        c.consume();
    }
  }
}
```

4. Look at the following program:

```
class ArrayList {
```

```
protected int[] array;
protected int next;

public void add(int i) {
  if( next==array.length ){
    resize( );
  }
  array[ next ] = i;
  next++;
}

public int[] getElems() {
      return array;
}

public void setElems(int[] ia){
      array = ia;
      next = ia.length;
}
```

```
protected void resize() {
  if( next==array.length ) {
    int[] oa = array;
    array = new int[2*oa.length];
    setElems( oa );
  }
}

public String toString() {
  if( array.length == 0 ) return "[]";
  StringBuffer buf =
    new StringBuffer("[ " + array[0]);

  for( int i=1; i < next; ++i ) {
    buf.append(", " + array[i]);
  }
  buf.append(" ]");
  return buf.toString();
}
```

```
}
```

a. What aliasing problems can arise in the example program?
b. Write example code for every problem.
c. Change the code of `ArrayList` in a way so that there are no more aliasing problems.
d. Annotate `ArrayList` with the ownership type modifiers