Points off	1	2	3	4	5	Total off	Net Score

CS 307 – Midterm 1 – Fall 2008

Your Name_________________________________

Your UTEID _______________________________

Circle yours TA's name: Mikie Sarah Ron

Instructions:

1. Please turn off or silence your cell phones
2. There are **5** questions on this test.
3. You will 2 hours to complete the test.
4. You may not use a calculator or any other electronic devices while taking this test.
5. When code is required, write Java code.
6. When writing a method, assume the preconditions of the method are met.
7. When writing a method you may add helper methods if you wish.
8. When you complete the test show the proctor your UTID and give them the test and any scratch paper. Please leave the room quietly.

1. (2 points each, 30 points total) Short answer questions. Place your answers on the attached answer sheet. For code sample state the output. If the code would cause a syntax error answer "syntax error". If it would cause an exception error answer "exception". If it would result in an infinite loop answer "infinite loop".

A. What is the output of the following client code?

```
// pre: list != null, list.length > 2
public void a(int[] list){
  list[0] += list[1] + list[2];
  list[1]++;
  list[2]--;
}

// client code
int[] data = {3, 1, 5};
a(data);
System.out.println( data[0] );
```

B. Would the following code, which calls method `a` from question A, result in a runtime error? Briefly explain your answer.

```
int[] moreData = {0, 0};
a(moreData);
```

Questions C through F make use of the following methods.

```
public int b(int x){
    x *= 2;
    return x;
}

public int b(int x, int y){
    x--;
    y++;
    return x * y;
}

public int c(int x){
    x /= 2;
    return x;
}
```

C. What is output by the following client code?

```
int x = 3;
b(x);
System.out.println(x);
```

D. What is output by the following client code?

```
int y = 3;
int z = c( b(y, y) );
System.out.println(z);
```

E. What is output by the following client code?

```
int m = 4;
int n = 3;
int p = b(m) + b(n, m);
System.out.println(p);
```

F. What is output by the following client code?

```
int[] list = {4, 2, 6};
list[0] = b(list[1], list[2]);
list[1] = c(list[1]);
System.out.println( Arrays.toString(list) ); // prints out elements
```

For questions G – O consider the following classes and interfaces.

```
public interface Taxable{
      public int rate();
}

public abstract class Item implements Taxable{
      private String unit;

      public Item(String s){
            unit = s;
      }

      public String baseUnit(){
            return unit;
      }

      public abstract int basePrice();

      public int cost(){
            int tax = (int)(basePrice() * (rate() / 100.0));
            return basePrice() + tax;
      }

      public String toString(){
            return "item";
      }
}

public class Shoes extends Item{

      private int price;
      private int taxRate;

      public Shoes(int p, int r){
            super("pair");
            price = p;
            taxRate = r;
      }

      public int basePrice(){
            return price;
      }

      public int rate(){
            return taxRate;
      }

      public void increase(){
            price++;
      }

      public String toString(){
            return "shoes";
      }
}
```

G. State if the following declarations are legal or illegal. 1 point each

```
Taxable t1 = new Taxable();  // 1
Taxable t2 = new Shoes(50, 5); // 2
```

H. State if the following declarations are legal or illegal. 1 point each

```
Object obj = new Shoes(50, 5);  // 1
Shoes s1 = new Object(); // 2
```

I. State if the following declarations are legal or illegal. 1 point each

```
Shoes s2 = new Shoes(50, 5); // 1
Item it = new Item("box"); // 2
```

J. What is the output of the following code?

```
Shoes sj = new Shoes(50, 5);
sj.price += 10;
System.out.println( sj.basePrice() );
```

K. What is the output of the following code?

```
Shoes sk = new Shoes(50, 5);
System.out.println( sk.rate() );
```

L. What is the output of the following code?

```
Item it = new Shoes(50, 5);
System.out.println( it );
```

M. What is the output of the following code when method `m2` is called?

```
public void m1(Shoes s){
     s.increase();
     s.increase();
}

public void m2(){
     Shoes sm = new Shoes(10, 10);
     m1(sm);
     System.out.println( sm.basePrice() );
}
```

N. What is the output of the following code when method `n2` is called?

```
public void n1(Shoes s){
      s.increase();
      s = new Shoes(100, 10);
      s.increase();
}

public void n2(){
      Shoes sn = new Shoes(24, 5);
      n1(sn);
      System.out.println( sn.basePrice() );
}
```

O. Consider the following class:

```
public class Peg{

      private String color;

      private Peg(String c){
           color = c;
      }

      public String getColor(){
           return color;
      }
}
```

Explain why the following class won't compile.

```
public class GamePeg extends Peg{

      private int size;

      public GamePeg(int s){
           size = s;
      }

      public int getSize(){
           return size;
      }
}
```

2. Arrays and Classes. (15 Points) In class we developed an `IntList` class to represent a list of `ints` in order to demonstrate encapsulation and the syntax for building a class in Java. As discussed in class the internal storage container may have extra capacity.

Complete a method for the `IntList` class, `subList`. This is an instance method in the `IntList` class that creates and returns a new `IntList`. The method specifies the starting position in the original list which will be the first element in the sub list and the number of elements in the sub list. The calling object is not changed in any way.

You may not use any other methods in the `IntList` class except the default constructor unless you define and implement them yourself. You may not use objects or methods from the Java `System`, `Arrays` or `ArrayList` classes.

Here is the method header:

```
// pre: 0 <= start < size(), num > 0, start + num <= size()
// this IntList is not changed in any way
public IntList subList(int start, int num){
```

Examples of calls to subList.

Original IntList	IntList returned by method
[0, 3, -2, 4, 5].subList(1,1)	[3]
[0, 3, -2, 4, 5].subList(1,2)	[3, -2]
[0, 3, -2, 4, 5].subList(0,3)	[0, 3, -2]
[0, 3, -2, 4, 5].subList(4,1)	[5]
[0, 3, -2, 4, 5].subList(0,5)	[0, 3, -2, 4, 5]

```
public class IntList{

     public static final int DEFAULT_SIZE = 10;

     private int[] container;
     private int listSize;

     public IntList(){
          container = new int[DEFAULT_SIZE];
          listSize = 0;
     }

     // pre: 0 <= start < size(), num > 0, start + num <= size()
     // this IntList is not changed in any way
     public IntList subList(int start, int num){
          // Complete this method on the next page.
// pre: 0 <= start < size(), num > 0, start + num <= size()
// this IntList is not changed in any way
public IntList subList(int start, int num){
     assert 0 <= start && start < size() && num > 0 &&
          (start + num) <= size() : "failed precondition";
```

3. Arrays and Objects. (20 points) Write a method that determines the average number of letters and digits per String given an array of Strings. Elements of the array may be null.

Use the following method from the Character class to determine if a character from a String is a letter or digit.

static boolean **isLetterOrDigit**(char ch) Determines if the specified character is a letter or digit.

You will also have to use the length and charAt methods from the String class.

Here is the method header:

```
// pre: words != null
// post: return the average number of letters and digits per String
// in words. If there are zero non-null elements in words
// return -1.0. null elements do not count as Strings when figuring
// the average number of letters and digits per String
public double averageLettersAndDigitsPerString(String[] words){
```

Here are some example results:

Given the array {null, null, null} the method would return -1.0.

Given the array {null, "cat", "hat", "..."} the method would return 2.0.

Given the array {"!!!!!", "?!?!?!?", ">>>>"} the method would return 0.0.

Given the array {"!!!!!", "?!3.....", ">>>>"} the method would return 0.3333333333333333.

Complete the method on the next page.

```
// pre: words != null
// post: return the average number of letters and digits per String
// in words. If there are zero non-null elements in words
// return -1.0. null elements do not count as Strings when figuring
// the average number of letters and digits per String
public double averageLettersAndDigitsPerString(String[] words){
     assert words != null : "failed precondition";
```

4. 2D arrays. (20 points) Complete a method that given a two dimensional array of `ints` alters the array so that any elements in the original array that are equal to `0` cause all the elements in the row and column that element is a part of to become `0`.

Here is an example. Given the following initial 2d array

2	1	4	3	10
3	2	1	2	5
1	3	0	2	2
12	5	5	1	-6
9	0	5	-1	-5
-1	4	5	7	9

The resulting 2d array would be:

2	0	0	3	10
3	0	0	2	5
0	0	0	0	0
12	0	0	1	-6
0	0	0	0	0
-1	0	0	7	9

If there are no `0`s in the original 2d array then the 2d array will be unchanged. You may only use native arrays and primitives in your solution.

Assume the preconditions of the method are met. Do not write code to check the precondition.

```
// pre: mat != null, mat.length > 0, mat[0].length > 0, mat is a
// rectangular 2d array
// post: alter mat as described in the question.
public void applyZeros(int[][] mat){
```

```
// more room on the next page if necessary
```

5. (Implementing classes, 15 points) Complete a class that models a Building.

A building has the following properties:

- A number of floors. This will be a whole number.
- The square feet per floor, a whole number. Every floor has the same number of square feet.
- The number of square feet in the building that are currently occupied. This will be a whole number. This will be less than or equal to the number of floors * the square feet per floor.

Complete a complete `Building` class that has the following properties.

- Instance variables to store all of the relevant information for a building as explained above.
- A constructor to initialize all values based on parameters.
- A method to return the percentage of the buildings square footage that is occupied. This will be a floating point number between 0.0 and 1.0.
- A method to increase or decrease the amount of occupied square feet in the building by some given amount.

Complete the class below. You may add other methods if you wish. State your preconditions for each method and use assert statements to check them.

```
// more room on the next page if necessary
```

Scratch paper

Question 1 answer Sheet

Name____

A. __________________________________

B. __________________________________

C. __________________________________

D. __________________________________

E. __________________________________

F. __________________________________

1.

2.

G. __________________________________

1.

2.

H. __________________________________

1.

2.

I. __________________________________

J. __________________________________

K. __________________________________

L. __________________________________

M. __________________________________

N. __________________________________

O. __________________________________

Scratch Paper

Scratch Paper

Scratch Paper

Scratch Paper

Scratch Paper