Department of Computer Science

Purdue University, West Lafayette

Fall 2011: CS 180 Problem Solving and OO Programming

Final Examination: Part A.

You may consult the textbook and your *hand written* notes.

Friday December 16, 2011 1:00-3:00 pm. LAMBF101.

Name:____

CS Login ID:____

Recitation Section:____

Make sure your CS Login ID appears at the top of each page.

- This exam has two parts. Part A has 10 multiple choice questions. Part B has two programming problems. Complete Part A, return to the proctor and get Part B.
- Use the Scantron sheets to write your answer. Make sure to write your Purdue ID (NOT Purdue Login ID) and your name on the Scantron answer sheet.
- Part A is worth 10 points and Part B worth 20 points.

Q 1 Consider the following statement sequence S1.

```
public class ArrayTest{
  public static void exchange(int [] a, int [] b){
        int [] temp;
        temp=a;   a=b; b=temp;
    }
    public static void  main(String [] arg){
        int [] x={1, 2, 3},  y={4, 5, 6};
        exchange(x, y);
    }
System.out.println(x[0]+"  "+y[0]);
}
```

Upon execution of S1

(a) 4 1 is printed

(b) 5 2 is printed

(c) an array index out of bounds exception is generated

(d) 1 4 is printed

Q 2 Consider the following method.

```
public int GCD(int x, int y){
 if(x==0)
      return y;
 else{
     return(GCD(y, x%y));
     }
}
```

When called as GCD(6, 4)

(a) it will return 1

(b) it will return 2

(c) it will return generate an arithmetic exception

(d) it will return 6

Q 3 Consider the following statement sequence S2:

```
int [] a={2, -2, 4};
int x=2, index=1;
boolean found=false;
while(index<a.length&&!found){
  if(x==a[index]){
    found=true;
    break;
  } // End of if
  index++;
}// End of while
  if(found){
    System.out.println("x is in a");
   }else{
    System.out.println("x is not in a");
  }// End of if
```

Execution of S2 will

(a) display: x is in a

(b) display: x is not in a

(c) display: 2

(d) result in a run time exception

Q 4 Consider the following statement sequence S3:

```
int [] a={1};
a[0]=5;
System.out.println(a[1]);
```

S3 will

(a) result in a run time exception

(b) display: 6

(c) result in a compile time error

(d) display: 5

Q 5 Consider the following statement sequence S4:

```
public void increment(int [] x){
  x[1]=x[1]+1;
}

public static void  main(String [] arg){
   int[] p={5};
   increment(p);
   System.out.println(p[0]);
}
```

Execution of S4 will

(a) result in a run time exception

(b) display: 6

(c) display: 5

(d) display: 1

Q 6 Consider the following statement sequence S5:

```
public static void test(int[]  x, int index){
  x[index]=x[index]+1;
}

public static void  main(String [] arg){
  int [] a={2, -2, 4};
  test(a, 1);
  System.out.println(a[1]);
```

S5 will

(a) display: 3

(b) result in a compile time error

(c) display: 5

(d) display: -1

Q 7 Consider the following declaration:

```
public class Mango extends Fruit implements FruitMarket, MyListener{
```

Based on the above declaration which one of the following conclusions is **incorrect**?

(a) FruitMarket is an interface

(b) MyListener is an interface

(c) Mango is an object

(d) Fruit is a class

Q 8 Class X implements ActionListener and MouseListener. Which one of the following statements is **incorrect**?

(a) X must implement the mouseExited method.

(b) X must implement the mousePressed method and the actionPerformed method.

(c) X must implement the actionPerformed method and all methods in the MouseListener interface.

(d) X must implement all methods in MouseListener but need not implement the actionPerformed method.

Q 9 Consider the following methods

```
public  static int find(int x, int y){return(1);}
public  static int find(int [] x, float [] y){return(2);}
public  static int find(int x[], float y){return(3);}
public  static int find(int x, float y) {return(4);}
```

Now consider the following statement sequence S6:

```
int[] a=new int[10]; int b=0;
int z=find(a, b);
System.out.println(z);
```

What will be displayed when S6 is executed?

(a) 4

(b) 3

(c) 2

(d) 1

Q 10 Variables x and y are declared in class Test as follows:

```
public class Test{
  private static int x;
  public int y;

  public static void main(String args[]){
    System.out.println(x+" "+y);
     }
}
```

Which one of the following statements is **incorrect** regarding the use of x and y in the program above?

(a) x can be used in main().

(b) x cannot be used in a class other than Test.

(c) y can be used in a class other than Test.

(d) y can be used in main().

<End of Part A of Final exam CS 180. Fall 2011.>

Department of Computer Science
Purdue University, West Lafayette

Fall 2011: CS 180 Problem Solving and OO Programming
Final Examination. Part B.
You may consult any book and your notes.
Friday December 16, 2011 1:00-3:00 pm. LAMBF101.

Name:____

CS Login ID:____

Recitation Section:____

Make sure your CS Login ID appears at the top of each page.

Ignore `import` statements. Write **ONLY** the part of the code you are asked to write.

Q 1 In this question you are required to write three methods. Do not write any `import` statements or class declarations. Simply write the method requested. A correctly written method is worth 3 points. There is no partial credit in this question for a method. All methods are `public` but not `static`.

(i) Method `rotate` takes a single dimensional array of characters named `myChars` as an input parameter. It rotates the contents of `myChars` to the right by one position. The method does not return any value, it simply changes the array that is input as a parameter. Following are two examples (the dash (-) is used below to indicate a space character).

Example 1:

myChar= [a b c - -]
myChar (after rotation)= [- a b c -]

Example 2

myChar= [9 5 * 3 c / p]
myChar(after rotation)= [p 9 5 * 3 c /]

Write the `rotate` *method on the following page.*

Name:______________________ Purdue Login ID: _______________

// Code for rotate() begins here

// Code for `rotate` ends here.

(ii) Consider the following recursive definition of a sequence `t`:

$$t_1 = 10$$
$$t_n = t_{n-1} - 3; \text{for } n > 1$$

Thus, $t_1 = 10$, $t_2 = 10 - 3 = 7$, $t_3 = 7 - 3 = 4$ and so on. Write a *recursive* method named `nextItem()` that takes integer $n > 0$ as input and returns the n^{th} integer in the sequence. For example, if $n = 4$ then the method should return 0 (because $t_4 = t_3 - 3 = 4 - 3 = 1$).

// Code for `nextItem()` begins here

// Code for `nextItem()` ends here.

(iii) Write a method named `simpleTimer()`. This method takes an integer d as input. It then prints a 1 on the console. Then, after a delay of d seconds, it prints 2. Then, again after a delay of d seconds it prints 3, and so on. The method does not return any value. It exits after $d*10$ seconds. Thus, for example, if $d = 6$ then the method exits after 60 seconds. If you need, you may use `Thread.sleep()` and/or `System.currentTimeMillis()` while coding this method.

// Code for `simpleTimer()`begins here

// Code for `simpleTimer()` ends here.

Q 2 You will be writing two classes: `Search` and `Test`. `Search` extends `Thread` and is used for searching for a given integer in a two dimensional array. `Test` uses objects of type `Search` to search for an integer.

Class Test: The `main()` method in this class performs the following tasks in the given sequence.

(a) Calls `createArray()` that returns a two dimensional array of integers. **You are not required to write `createArray()`** or even call it; the code to call this method is already added.

No credit

(b) Creates two objects named `s1` and `s2` of type `Search`. Each object takes array `x` and three integers `num`, `start`, and `end` as inputs, where `num` is the number to be searched in `x`. The number is searched among elements of `x` in rows starting at `start` and ending at `end`. For `s1`, `start`=1 and `end`=n/2 and for `s2`, `start`=n/2+1 and `end`=n.

1 point

(c) Starts threads `s1` and `s2`.

1 point

(d) Waits for `s1` and `s2` to join.

1 point

(e) Calls `getResult()` in `s1` and `s2` to obtain the outcome of search. `getResult()` returns true if the given number exists in the array and false otherwise.

1 point

(f) Uses the results obtained from the two objects `s1` and `s2` to determine whether or not `num` is found in `x`. It then prints true if it exists and false otherwise.

1 point

Class Search: This class extends `Thread`.

The constructor saves the parameters supplied by `main()`.

1 point

The `run()` method searches for the given integer in the given array but only among the elements in rows given by `main`. For example, if the start row is 1 and the end row is 5 then the search is only among the array elements in rows 1, 2, 3, 4, and 5. A private boolean variable `found` is set to true if the search is successful otherwise it is set to false.

5 points

The `getResult()` method returns `found`. **You are not required to write this method.**

No credit

```
// Code for class Test begins here.

public class Test{

 final int n=10; // Your program must work for any  n>0.
 int [][] x=new int[n][n]; // Create a 2-dimensional array of integers.

 public static void main(String [] args){

 x=createArray(n); // Create x
// Write below the remainder of the code.
```

```
}\\ End of main()
```

```
// Code for class Search begins here.

public class Search{
 private int [][] x; // A 2-dimensional array of integers.
 private int num; // Holds the number to be searched.
 private int start, end; // Staring and ending rows in x;
 private boolean found; // Outcome of search.

 public  Search(){ // Write code for the constructor.










}// End of constructor

public void run(){ // Write code for run();
// continue on the next page if needed.
}// End of run()
```

<End of Part B of Final exam CS 180. Fall 2011.>