Name and, if possible, ID#:

AMERICAN UNIVERSITY OF ARMENIA

College of Science and Engineering

COMP120 Introduction to Object-Oriented Programming

FINAL EXAM

8/15

Date:	Monday, May 18 2015
Starting time:	09:20
Duration:	1 hour 40 minutes
Attention:	**ANY TYPE OF COMMUNICATION IS PROHIBITED**

Please write down your name at the top of all used pages

Problem 1

Consider below a ***public interface Valuable*** that includes the only method ***public double value(double x)***:

```
public interface Valuable {

      public double value(double x);
}
```

1.1 Implement a ***public class Function*** that encapsulates a member variable of type ***Valuable*** and computes its integral in the specified range from x_1 to x_2 using the approximation:

$$\int_{x1}^{x2} f(x)dx \approx \frac{x_2 - x_1}{6}\left(f(x_1) + 4f\left(\frac{x_1 + x_2}{2}\right) + f(x_2)\right)$$

```
public class Function {

      private Valuable f;
      private double dx;

      public Function(Valuable newValuable, double newDX) {
            //TO BE IMPLEMENTED
      }

      public double integral(double x1, double x2) {
            //TO BE IMPLEMENTED
      }
```

1.2 Implement an expression

$$\sqrt{x^2 + a} + \sqrt{x^2 + b}$$

as a ***public class Roots*** that implements the interface ***Valuable*** and encapsulates double parameters ***a*** and ***b***. The parameters are initialized by the two-argument constructor ***public Roots(double newA, double newB)***;

1.3 In a separate ***public static void main(String args[])*** write a code that inputs two double values, creates an object of type ***Roots*** and, using the class ***Function***, prints the value of its integral from $x_1 = 1.0$ to $x_2 = 2.0$:

```
public static void main(String args[]) {
      Scanner input = new Scanner(System.in);
      double a = input.nextDouble(), b = input.nextDouble();

      //TO BE COMPLETED

}
```

3

Use the backside, if needed

OOP.FT.180515.H094

Problem 1

1.2

```
public class Roots{
    private double a;
    private double b;

public Roots(double newA, double newB){ implements Valuable {
    a = newA;      public Object[] objs = {newA, newB};
    b = newB;      return objs;
}
```

((1))

1.3

```
Roots Obj1 = new Roots();
Function Obj2 = new Function();

Obj2.integral(1.0, 2.0);
Obj1.calculate();
```

((1))

```
public double calculate(
    double gnum = Math.sqrt(Math.pow(x, 2) + a);
    return gnum;
```

Problem 2

```
public void drawCap(Graphics g){
    g.drawOval( size/6, -size/6, size/3, size/3 );
    g.setColor(Color.white);
    g.fillOval( size/6+1, -(size/6-1), 2*(size-2)/3, (size-2)/3 );
```

OOP.FT.180515.HOGY

Name and, if possible, ID#:

```
public class Life extends Animator {

      private boolean grid[][] = new boolean[100][100];
      private int cellSize = 4;

      public void init() {
            for (int row = 0; row < grid.length; row++)
                  for (int col = 0; col < grid[0].length; col++)
                        grid[row][col] = Math.random() < 0.5;
      }

      private int sum9(int row, int col) {
            int result = grid[row][col] ? -1 : 0;

            for (int i = Math.max(0, row - 1);
                        i < Math.min(grid.length - 1, row + 1); i++)
                  for (int j = Math.max(0, col - 1);
                              j < Math.min(grid[0].length - 1, col+ 1); j++)
                        result += grid[i][j] ? 1 : 0;

            return result;
      }

      public boolean tick() {
            //TO BE IMPLEMENTED
      }

      public void snapshot(Graphics g) {
            //TO BE IMPLEMENTED
      }
}
```

(1)

```
if(result < 2 || result > 3 && grid[row][col] == true){
      return false;
}
else if(result == 2 || result == 3 && grid[row][col] == true){
      return true;
}
else if(result == 3 && grid[row][col] == false){
      return true;
}
```

(2)

```
if (true){
      g.fillRect(0, 0, getWidth(), getHeight());
} else { g.drawRect(0, 0, getWidth(), getHeight()); }
```

~~no need to check if it is false because all the squares are~~

3

Use the backside, if needed

OOP.RT.180515.H094