

Name and, if possible, ID#:

✓
AMERICAN UNIVERSITY OF ARMENIA
College of Science and Engineering
COMP120 Introduction to Object-Oriented Programming

FINAL EXAM

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

13/15

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 derivative at the specified point *x* using the approximation:

$$f'(x) \approx \frac{f(x+dx) - f(x-dx)}{(2 * dx)}$$

```
public class Function {  
    private Valuable f;  
    private double dx;  
  
    public Function(Valuable newValuable, double newDX) {  
        //TO BE IMPLEMENTED  
    }  
  
    public double derivative(double x) {  
        //TO BE IMPLEMENTED  
    }  
}
```

- 1.2 Implement an expression

$$\exp(-a * (x - c)^2)$$

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

- 1.3 In a separate *public static void main(String args[])* write a code that inputs two double values, creates an object of type *Gauss* and, using the class *Function*, prints the value of its derivative at the *x = 1.0* point:

```
public static void main(String args[]) {  
    Scanner input = new Scanner(System.in);  
    double a = input.nextDouble(), c = input.nextDouble();  
  
    //TO BE COMPLETED  
}
```

Use the backside, if needed

Page 1 of 4

OOP FT. 1805PS-H100

Ex 8

```
public class Function {  
    private Valuable f;  
    private double dx;
```

```
public Function (Valuable newValuable, double newDX)  
{  
    f = newValuable;  
    dx = new DX;    }
```

```
public double derivative (double x) {  
    double fd = (f.value(x+dx) - f.value(x-dx)) / (2*dx);  
    return fd;    }
```

```
public interface Valuable {  
    public double value (double x);  
}
```



```
public class Gauss implements Valuable {  
    private double a;  
    private double c;
```

```
    public Gauss(double newA, double newC) {  
        a = newA;  
        c = newC;
```

```
    }
```

```
    public double value(double x) {
```

```
        double value =  $\exp(-a * (x - c) * (x - c))$ ;
```

```
        return value;
```

```
    }
```


3]

```
public static void main (String args []) {
    Scanner input = new Scanner (System.in);
    double a = input.next Double ();
    double c = input.next Double ();
```

```
    Valueable uass g = new Gauss (a, c);
```

```
    Function funet = new Function (g, 2, 5);
    System.out.print (funet.derivative (x));
```

5

3


```
Public class Rook extends ChessPiece {  
    Private Rectangle cap;
```

```
    public Rook(int size) {  
        super(size);
```

```
        cap = new Rectangle(size/3, size/3);  
    }
```

@Override

```
    public void drawCap(Graphics g) {
```

```
        g.drawRect(size/2 - size/6, size - size/6,  
                    cap.width, cap.height);  
    }
```

}

4


```

public class ... {
    private boolean grid [][] = new boolean [100, 100],
    private int cellSize = 4;

```

```

    public boolean tick() {
        for (int i = 0; i < grid.length; i++)
            for (int j = 0; j < grid[0].length; j++)
                if (sum(i, j) < 2 || sum(i, j) > 3)
                    boolean t = false;
                    if (sum(i, j) == 2 || sum(i, j) == 3)
                        t = true;
                    if (sum(i, j) == 3)
                        t = true;

```

```

    public void snapshot (Graphics g) {

```

```

        g.clearRect(0, 0, width, height);
        paint (new Graphics);

```

```

        boolean t = tick();

```

```

        if (t == true) {

```

```

            paint (new Graphics);

```

```

        }

```

```

    }

```

draw ?