***Quiz 1  
Instructor: Albert Hambardzumyan***

***Duration: 2h***

***Classes***

***1 (20 points).*** Implement the concept of point in the given below ways by including the following methods:   
The constructor that takes **double** new\_x, **double** new\_y;

**double** get\_x(); // x coordinate  
 **double** get\_y(); // y coordinate  
 **int** int\_x(); // returns rounded x coordinate   
 **int** int\_y();// returns rounded x coordinate  
 **void** shift(double x, double y); // shifts own x, and y by the given arguments  
 **double** distance(MyPoint p); // calculates the distance between given point and itself

Write short test for each of your implementation.

***2 (80 points).*** Implement the concept of line in the given below ways by including the following methods:

The constructor that takes **double** x1, **double** y1, **double** x2, **double** y2;

The constructor that takes MyPoint p1, MyPoint p2;

**double** get\_x(); //returns starting x coordinate

**double** get\_y(); //returns starting y coordinate

**int** int\_x(); //returns rounded starting x coordinate

**int** int\_y(); //returns rounded starting y coordinate

**double** end\_x(); //returns ending x coordinate

**double** end\_y(); //returns ending y coordinate

**int** end\_int\_x(); //returns rounded ending x coordinate

**int** end\_int\_y(); //returns rounded ending y coordinate

**double** length(); //returns length

**double** angle(); //returns angle relative to x-axis

**void** shift(**double** dx, **double** dy); //shifts the line by dx and dy

**void** rotate(**double** da); //rotates by da around starting point

1.1

**public class** MyLine1 {

**private** MyPoint start;

**private double** len, ang;

}

1.2

**public class** MyLine2 {

**private** MyPoint start, end;

}

1.3

**public class** MyLine3 {

**private** MyPoint ends[];

}

1.4

**public class** MyLine0 **extends** MyPoint {

**private double** len, ang;

}