

FONDAMENTI DI INFORMATICA II (Prof. E. Lamma) – 17 settembre 2009
PROVA PRATICA – LINGUAGGIO JAVA

NOTA PER LA CONSEGNA DEL COMPITO

Il compito deve essere stato **compilato ed eseguito**.

E' importante che tutti i file e le classi si chiamino come richiesto.

Nel testo .java (come commento) va indicato il nome dello studente.

Esercizio (da realizzare in un UNICO FILE .java)

Nella soluzione, prediligere il maggior riutilizzo di codice e la maggiore protezione possibile.

- Si realizzi una classe astratta **Distance** che rappresenta la generica misura di distanza tra due punti sul piano cartesiano. Tale classe ha come attributi **p1** e **p2** (di tipo java.awt.Point), che rappresentano due punti nello spazio cartesiano. La classe definisce il metodo **abstract public double getDistance()**; e codifica un metodo costruttore a 2 argomenti (p1 e p2, i due punti cartesiani).
- Si realizzi poi un componente software **EuclideanDistance**, che deriva da **Distance**. Tale componente codifica un metodo costruttore a 2 argomenti (p1 e p2: due punti cartesiani) e implementa il metodo **public double getDistance()**. Il metodo restituisce un valore di tipo **double** che rappresenta la distanza euclidea tra i punti p1 e p2.

NOTA: Per due punti in due dimensioni, $P = (p_x, p_y)$ e $Q = (q_x, q_y)$, la distanza è calcolata come:

$$\sqrt{(p_x - q_x)^2 + (p_y - q_y)^2}$$

- Si realizzi poi un metodo **main** in una classe **Prova** che:
 1. Crei 3 oggetti **puntoA**, **puntoB** e **puntoC** istanze della classe **Point**, con attributi come specificato in tabella:

	x	y
puntoA	2	1
puntoB	4	4
puntoC	6	3

2. Crei 2 oggetti **distance1** e **distance2** istanze della classe **EuclideanDistance**. **distance1** è creato utilizzando **puntoA** e **puntoB** come parametri, **distance2** utilizza invece **puntoB** e **puntoC**.
3. Calcoli e stampi a video, utilizzando il metodo getDistance() su **distance1** e **distance2**, il valore della distanza euclidea tra puntoA e puntoB e tra puntoB e puntoC.

Esempio di output:

Distanza 1: 3.605551275463989

Distanza 2: 2.23606797749979

Nota: La classe java.awt.Point, che deve essere importata, è consultabile sulle API di JAVA. Uno dei suoi costruttori è Point(int x, int y).E' possibile accedere ai valori di x e di y utilizzando i metodi getX() e getY().

```
Point p = new Point(3, 2);
```

```
//crea un oggetto p istanza di Point che rappresenta un punto in (x,y) = (3, 2)
```

```
double valoreX = p.getX();
```

```
//valoreX vale 3.0
```

```

import java.awt.Point;

abstract class Distance {
    protected Point p1;
    protected Point p2;

    public Distance(Point p1, Point p2) {
        this.p1 = p1;
        this.p2 = p2;
    }

    public abstract double getDistance();
}

class EuclideanDistance extends Distance {

    public EuclideanDistance(Point p1, Point p2) {
        super(p1, p2);
    }

    public double getDistance() {
        double dx = Math.pow((p1.getX()-p2.getX()), 2);
        double dy = Math.pow((p1.getY()-p2.getY()), 2);
        return Math.sqrt(dx+dy);
    }
}

public class Prova {

    public static void main(String[] args) {
        Point puntoA = new Point(2, 1);
        Point puntoB = new Point(4, 4);
        Point puntoC = new Point(6, 3);

        EuclideanDistance distance1 = new EuclideanDistance(puntoA, puntoB);
        EuclideanDistance distance2 = new EuclideanDistance(puntoB, puntoC);

        System.out.println("Distanza 1: " + distance1.getDistance());
        System.out.println("Distanza 2: " + distance2.getDistance());
    }
}

```