

```
1 /**
2  * Classe che simula un poligono
3  * @author radaelli11353
4  */
5 public class Polygon {
6     public final int MAX_DIM;
7     private double[] x;
8     private double[] y;
9     private int actualDim;
10
11     /**
12     * Costruttore
13     * @param dim Dimensione massima dell'array
14     * @param x0 X del primo punto
15     * @param y0 Y del primo punto
16     */
17     public Polygon(int dim, double x0, double y0) {
18         MAX_DIM = dim;
19         x = new double[MAX_DIM];
20         y = new double[MAX_DIM];
21         actualDim = 0;
22         this.add(x0, y0);
23     }
24
25     /**
26     * Metodo che aggiunge un punto al poligono
27     * @param xn X del nuovo punto da inserire
28     * @param yn Y del nuovo punto da inserire
29     */
30     public void add(double xn, double yn) {
31         if(actualDim >= MAX_DIM) throw new IllegalArgumentException("Il numero massimo di
32         vertici è stato raggiunto (" + MAX_DIM + ")");
33         x[actualDim] = xn;
34         y[actualDim] = yn;
35         actualDim++;
36     }
37
38     /**
39     * Metodo che calcola il perimetro del poligono
40     * @return Perimetro del poligono
41     */
42     public double perimeter() {
43         double dx, dy;
44         double perimeter = 0;
45         for(int i = 0; i < actualDim; i++) {
46             if(i != actualDim - 1) {
47                 dx = x[i] - x[i+1];
48                 dy = y[i] - y[i+1];
49             } else {
50                 dx = x[i] - x[0];
51                 dy = y[i] - y[0];
52             }
53             perimeter += Math.sqrt(dx*dx + dy*dy);
54         }
55         return perimeter;
56     }
57
58     /**
59     * Metodo che restituisce l'area del poligono
60     * @return Area del poligono
```

```
61     */
62     public double area() {
63         double sum = 0;
64         for(int i = 0; i < actualDim - 1; i++) {
65             sum += x[i] * y[i+1];
66         }
67         sum += x[actualDim - 1] * y[0];
68         for(int i = 0; i < actualDim - 1; i++) {
69             sum -= y[i] * x[i+1];
70         }
71         sum -= y[actualDim - 1] * x[0];
72
73         return sum/2;
74     }
75
76     /**
77     * Metodo toString della classe
78     * @return Stringa contenente i vertici del poligono
79     */
80     @Override
81     public String toString() {
82         String text = "Punti: ";
83         for(int i = 0; i < actualDim; i++) {
84             text += x[i] + "," + y[i] + "; ";
85         }
86         return text;
87     }
88 }
```