

Esercizio 2.1 - Rettangolo.java

Testo:

Si costruisce un rettangolo partendo da una base, un'altezza e dalle coordinate del piano.

6

Consigli:

Per chi si affaccia per la prima volta al panorama della programmazione d'informatica, può risultare difficile la risoluzione di tale problema. Si cerchi, quindi, di comprendere ogni passaggio:

- costruttore: ne sono presenti due. Il primo crea un rettangolo con le dimensioni definite, il secondo, invece, permette all'utente di assegnare i valori

- metodi: possono essere suddivisi in più categorie. I metodi contenenti la dicitura "get" (`getBase`, `getAltezza`...) restituiscono il valore della dimensione richiesta o eseguono un'operazione specifica (`getArea`). Quelli contenenti il prefisso "set", invece, permettono di assegnare un nuovo valore alla variabile d'istanza.

```
public class Rettangolo
{
    private int b;
    private int h;
    private int x;
    private int y;
    // si costruisce un rettangolo con i parametri predefiniti
    public Rettangolo()
    {
        b = 1;
        h = 1;
        x = 0;
        y = 0;
    }
}
```

```

/* si costruisce un rettangolo con i parametri acquisiti dall'esterno
@param base la base del rettangolo
@param altezza l'altezza del rettangolo
@param ascissa l'ascissa del rettangolo
@param ordinata l'ordinata del rettangolo */
public Rettangolo(int base, int altezza, int ascissa, int ordinata)
{
    b = base;
    h = altezza;
    x = ascissa;
    y = ordinata;
}
/* si acquisisce la base
@return la base del rettangolo */
public int getBase()
{
    return b;
}
/* si acquisisce l'altezza
@return l'altezza del rettangolo */
public int getAltezza()
{
    return h;
}
/* si acquisisce l'ascissa
@return l'ascissa del rettangolo */
public int getAscissa()
{
    return x;
}

```

```

/* si acquisisce l'ordinata
@return l'ordinata del rettangolo */
public int getOrdinata()
{
    return y;
}
/* si modifica la base
@param nuovaBase la nuova misura della base*/
public void setBase(int nuovaBase)
{
    b = nuovaBase;
}
/* si modifica l'altezza
@param nuovaAltezza la nuova misura dell'altezza*/
public void setAltezza(int nuovaAltezza)
{
    h = nuovaAltezza;
}
/* si modifica l'ascissa
@param nuovaAscissa la nuova ascissa*/
public void setAscissa(int nuovaAscissa)
{
    x = nuovaAscissa;
}
/* si modifica l'ordinata
@param nuovaOrdinata la nuova ordinata*/
public void setOrdinata(int nuovaOrdinata)
{
    y = nuovaOrdinata;
}

```

```

/* si traslano le coordinate nel piano
@param trX lo spostamento in ascissa
@param trY lo spostamento in ordinata*/
public void traslazione(int trX, int trY)
{
    x = x + trX;
    y = y + trY;
}
/* si calcola il perimetro
@return il perimetro */
public int getPerimetro()
{
    return (b + h)*2;
}
/* si calcola l'area
@return l'area */
public int getArea()
{
    return b * h;
}
}

```

```

RettangoloTester.java
public class RettangoloTester
{
    public static void main(String[] args)
    {
        Rettangolo r = new Rettangolo();
        System.out.println("Perimetro: " + r.getPerimetro());
        System.out.println("Area: " + r.getArea());
        System.out.println("Expected perimetro: 4");
        System.out.println("Expected area: 1");
        Rettangolo rr = new Rettangolo(5, 3, 9, 2);
        System.out.println("Perimetro: " + rr.getPerimetro());
        System.out.println("Area: " + rr.getArea());
        System.out.println("Expected perimetro: 16");
        System.out.println("Expected area: 15");
    }
}

```