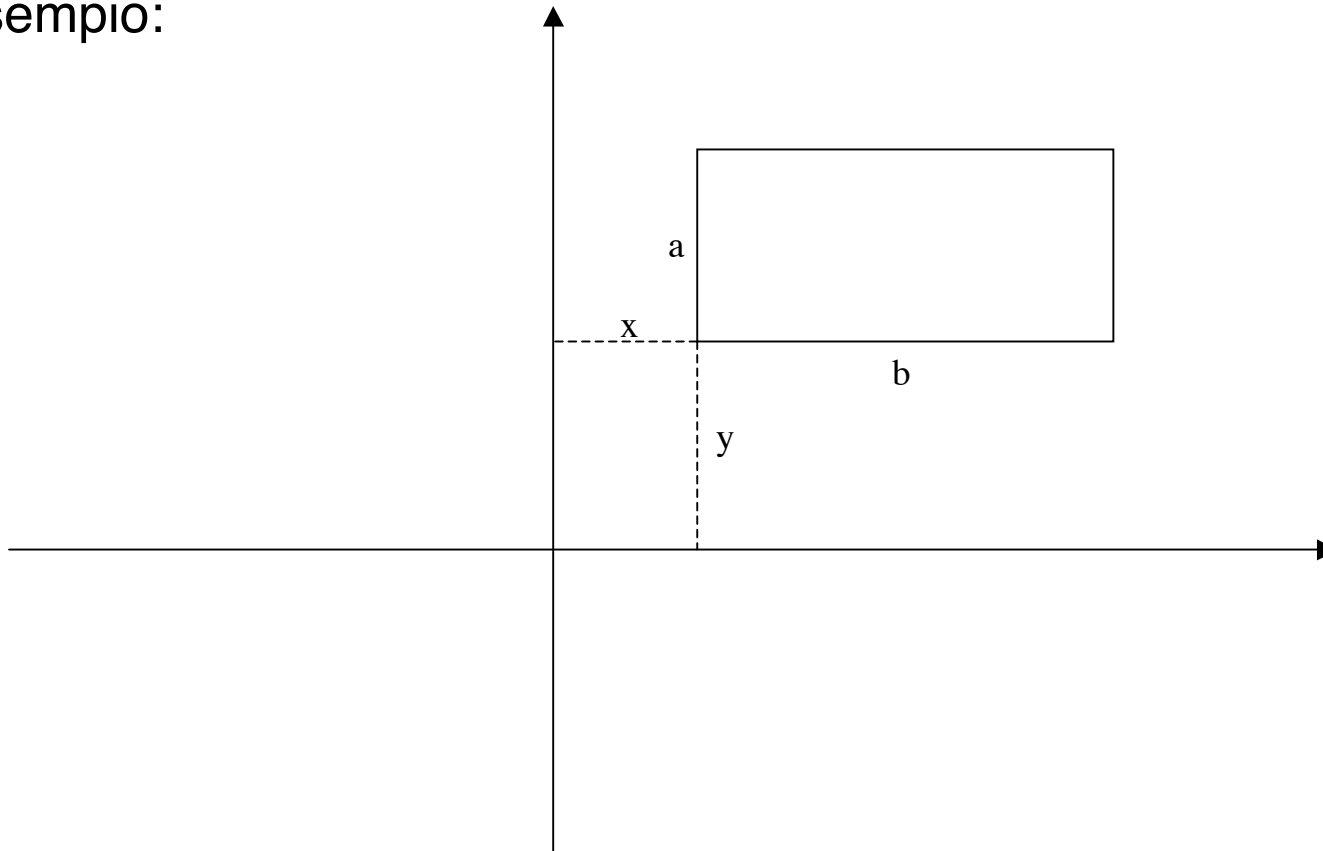


Esercizio 1

Si consideri un rettangolo nel piano cartesiano, rappresentato dalle sue coordinate (x,y) , dalla base (b) e altezza (a) . Ad esempio:



Esercizio 1

Si vuole realizzare una classe che rappresenta un rettangolo, memorizzando i suoi dati (x,y,a,b) e mettendo a disposizione dei metodi per:

- Calcolare il perimetro e l'area del rettangolo
- Determinare se il rettangolo è in realtà un quadrato
- Ruotare il rettangolo di 90 gradi a sinistra o a destra
- Determinare se due rettangoli si intersecano
- Determinare il minimo rettangolo che contiene due rettangoli

Si richiede quindi di completare la seguente classe:

Classe Rettangolo (1)

```
public class Rettangolo {
    public int x;    // ascissa del rettangolo
    public int y;    // ordinata del rettangolo
    public int base;
    public int altezza;
    // post: costruisce un rettangolo secondo i parametri di input
    public Rettangolo(int x, int y, int b, int a) {...}

    // post: costruisce un rettangolo secondo i parametri di input e
    //        posizionato sull'origine del piano cartesiano
    public Rettangolo(int b, int a) {...}

    // post: calcola il perimetro del rettangolo
    public int perimetro() {...}

    // post: calcola l'area del rettangolo
    public int area() {...}

    // post: ritorna true se il rettangolo e' un quadrato, false altrimenti
    public boolean quadrato() {...}

    // post: ruota il rettangolo di 90 gradi a sinistra
    public void ruotasx() {...}
}
```

Classe Rettangolo (2)

```
// post: ruota il rettangolo di 90 gradi a destra
public void ruotadx() {...}

// pre: r diverso da null
// post: ritorna true se r e "questo" rettangolo hanno intersezione
//       non vuota
public boolean intersezione(Rettangolo r) {...}

// pre: r diverso da null
// post: ritorna il piu' piccolo rettangolo che contiene r e
//       "questo" rettangolo
public Rettangolo contiene(Rettangolo r) {...}

public static void main(String[] args) {
// usare il mail per testare tutti i metodi della classe
}
}
```