

Programação orientada a objetos - Tema de casa

Problema 1

Preencha os métodos da classe Line para aceitar coordenadas como um par de tuplas e retornar a inclinação e a distância da linha.

In [3]:

```
class Line(object):

    def __init__(self,coor1,coor2):
        self.coor1 = coor1
        self.coor2 = coor2

    def distance(self):
        x1,y1 = self.coor1
        x2,y2 = self.coor2

        return ((x2-x1)**2 + (y2-y1)**2) ** 0.5

    def slope(self):
        x1,y1 = self.coor1
        x2,y2 = self.coor2

        return float((y2-y1)/(x2-x1))
```

In [4]:

```
# Output exemplo

coordinate1 = (3,2)
coordinate2 = (8,10)

li = Line(coordinate1,coordinate2)
```

In [5]:

```
li.distance()
```

Out[5]:

9.433981132056603

In [6]:

```
li.slope()
```

Out[6]:

1.6

Problema 2

Preencha a classe

In [23]:

```
class Cylinder(object):

    pi = 3.14

    def __init__(self,height=1,radius=1):
        self.height = height
        self.radius = radius
```

```
self.radius = radius

def volume(self):
    return Cylinder.pi * (self.radius ** 2) * self.height

def surface_area(self):
    top = Cylinder.pi * self.radius ** 2
    return top * 2 + 2 * Cylinder.pi * self.radius * self.height
```

In [24]:

```
# Exemplo de saída
c = Cylinder(2,3)
```

In [25]:

```
c.volume()
```

Out[25]:

56.52

In [26]:

```
c.surface_area()
```

Out[26]:

94.2

In []: