

# Untitled

November 23, 2018

```
In [2]: from IPython.display import Image
        from IPython.display import HTML
        import matplotlib.pyplot as plt
        import numpy as np
        import numpy.random as npr
        import math
```

Definir função para representar os sistemas lineares dispostos na atividade!

$$3x + 2y + 4z = 1$$

$$x + y + 2z = 2$$

$$4x + 3y - 2z = 3$$

```
In [55]: def resolucao():
        A = np.array([ [3, 2, 4], [1, 1, 2], [4, 3, -2] ])
        b = np.array([ 1, 2, 3 ])
        resultado = np.linalg.solve(A, b)

        return resultado
```

Função responsável por retorna um número de ponto flutuante, dentro dos limites estabelecidos!

```
In [53]: def Gerador_Aleatorio():
        x = round(random.uniform(-10, 10), 1)
        y = round(random.uniform(-10, 10), 1)
        z = round(random.uniform(-10, 10), 1)

        return x, y, z
```

Retorna a média da Matriz A

```
In [54]: def sistema_linear(x, y, z):
        equacao1 = (3 * x) + (2 * y) + (4 * z)
        equacao2 = x + y + (2 * z)
        equacao3 = (4 * x) + (3 * y) - (2 * z)

        return equacao1, equacao2, equacao3
```

```

In [68]: def calcular_media(equacao1, equacao2, equacao3):
    m_equacao1 = int(abs((1 - equacao1)/1)*100)
    m_equacao2 = int(abs((2 - equacao2)/2)*100)
    m_equacao3 = int(abs((3 - equacao3)/2)*100)

    return np.array([[m_equacao1, m_equacao2, m_equacao3]])

In [85]: def calcula_equacao():
    vet_media = np.array([[0, 0, 0]])

    for i in range(0,50):
        x, y, z = Gerador_Aleatorio()
        equacao1, equacao2, equacao3 = sistema_linear(x, y, z)

        media = calcular_media(equacao1, equacao2, equacao3)

        vet_media = np.append(vet_media, media, axis=0)

    print (vet_media)
    return vet_media

In [93]: med = calcula_equacao()

    percentuais = []

    for i in range(0, 50):
        aux = round(sum(med[i])/3,2)
        percentuais.append(aux)

    percentuais = np.array(percentuais)
    percentuais.sort()
    percentuais[25::] = 0

    print("Menor percentual: {}".format( min(percentuais) ))
    print("Maior percentual: {}".format( max(percentuais) ))

[[ 0  0  0]
 [1789 179 3035]
 [4150 1070 1025]
 [1420 405 1755]
 [1999 219 939]
 [ 50 225 1600]
 [1830 710 1255]
 [3130 770 1025]
 [5070 1185 2400]
 [6000 1210 2250]
 [1380 305 1115]
 [3370 705 589]

```

[2369 810 1125]  
[7980 1835 2225]  
[2330 750 1725]  
[2720 389 2829]  
[ 140 100 569]  
[3440 835 1834]  
[5160 1170 389]  
[5090 980 1764]  
[7400 1739 1640]  
[3760 830 2030]  
[3579 940 470]  
[ 190 350 2194]  
[4740 1015 865]  
[ 850 370 244]  
[1710 455 1550]  
[4250 944 990]  
[2850 560 3185]  
[3709 740 44]  
[3940 735 94]  
[6709 1515 1070]  
[1380 600 1670]  
[2410 869 605]  
[ 560 185 1584]  
[2810 565 630]  
[1370 160 2365]  
[3840 775 735]  
[3010 775 1960]  
[7250 1640 2945]  
[2440 735 1165]  
[2970 900 655]  
[1010 200 944]  
[1550 405 580]  
[4220 975 1805]  
[2000 345 1614]  
[4079 885 1005]  
[1139 365 1465]  
[7750 1614 1610]  
[2039 665 715]  
[2949 565 1720]]

Menor percentual: 0.0%

Maior percentual: 1554.67%