

▼ SEGUNDA CLASE COMPUTACIÓN ESTADÍSTICA

Angie Ximena Garzon - 1007749757

Llevar a código funcional las siguientes funciones:

$$1. Y_1 = \log(x)$$

$$2. Y_2 = \log(\sqrt{x})$$

$$3. Y_3 = \log(\sqrt{x^2 - 1})$$

$$4. Y_4 = \log \frac{\log(\sqrt{x-1})}{x^2+1}$$

$$5. Y_5 = \frac{x-1}{\log(x)}$$

$$6. Y_6 = \frac{1}{x} + \sqrt{x-1}$$

$$7. Y_7 = \frac{1}{x} + \sqrt{x-1} - \frac{1}{\log(\frac{1}{x+1})}$$

$$8. Y_8 = \frac{2}{x^2+1}$$

$$9. Y_9 = \frac{1}{n} \sum_{i=1}^n x_i$$

$$10. Y_{10} = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2$$

$$11. Y_{11} = \frac{\sqrt{y_{10}}}{y_9}$$

```
1
2 import math
3 from math import log
4
5 x1 = float(input('Ingrese un valor X: '))
6 if x1>0:
7     y1 = log(x1, 10)
8     print(f'Y es: {y1}')
9 else:
10    print('Fuera del dominio de y')
```

Ingrese un valor X: -1
Fuera del dominio de y

```
1 import math
2 from math import log
```

```

3 from math import sqrt
4
5 x2 = float(input('Ingrese un valor X: '))
6 if x2>0:
7     y2 = log(sqrt(x),10)
8     print(f'Y es: {y2}')
9 else:
10    print('Fuera del dominio de y')

```

Ingrese un valor X: -2
Fuera del dominio de y

```

1 import math
2 from math import log
3 from math import sqrt
4
5 x3 = float(input('Ingrese un valor X: '))
6 if (x3>1) or (x3<-1):
7     y3 = log(sqrt(x3**2 -1),10)
8     print(f'Y es: {y3}')
9 else:
10    print('Fuera del dominio de y')

```

Ingrese un valor X: 2
Y es: 0.23856062735983116

```

1 import math
2 from math import log
3 from math import sqrt
4
5 x4 = float(input('Ingrese un valor X: '))
6 if (x4>1):
7     y4 = log((sqrt(x4-1)),10) / (x4**2 +1)
8     print(f'Y es: {y4}')
9 else:
10    print('Fuera del dominio de y')

```

Ingrese un valor X: 10
Y es: 0.004723972819006559

```

1 import math
2 from math import log
3 from math import sqrt
4
5 x5 = float(input('Ingrese un valor X: '))
6 if (0<x5<1) or (x5>1):
7     y5 = (x5-1) / log(x5,10)
8     print(f'Y es: {y5}')
9 else:

```

```
10 print('Fuera del dominio de y')
11
```

Ingrese un valor X: 5
Y es: 5.7227062322935724

```
1 import math
2 from math import log
3 from math import sqrt
4
5 x6 = float(input('Ingrese un valor X: '))
6 if (x6>=1):
7     y6 = (1/x6) +sqrt(x6-1)
8     print(f'Y es: {y6}')
9 else:
10    print('Fuera del dominio de y')
```

Ingrese un valor X: 2
Y es: 1.5

```
1 import math
2 from math import log
3 from math import sqrt
4
5 x7 = float(input('Ingrese un valor X: '))
6 if (x7>=1):
7     y7 = (1/x7) + sqrt(x7-1) - log((1)/(x7+1))
8     print(f'Y es: {y7}')
9 else:
10    print('Fuera del dominio de y')
```

Ingrese un valor X: 3
Y es: 3.1338412568263188

```
1 import math
2 from math import log
3 from math import sqrt
4
5 x8 = float(input('Ingrese un valor X: '))
6 y8 = 2 / (x8**2 +1)
7 print(f'Y es: {y8}')
```

Ingrese un valor X: 3
Y es: 0.2

```
1 #9 -10.
2 import statistics
3
4 nums = []
5 print('¿Cuantos numeros quiere ingresar?')
```

```

6 n = int(input())
7 i = 0
8 while i < n:
9     print('Valor numero:', i+1)
10    var = float(input())
11    nums.append(var)
12    i+=1
13
14 prom = statistics.mean(nums)
15 print('El promedio sera: ', prom)
16
17 vari = statistics.variance(nums)
18 print('La varianza sera: ', vari)

```

```

¿Cuantos numeros quiere ingresar?
2
Valor numero: 1
56
Valor numero: 2
455
El promedio sera:  255.5
La varianza sera:  79600.5

```

```

1 # 11.
2 import math
3 from math import sqrt
4
5 y11 = sqrt(vari) / prom
6 print('Y11 es:', y11)
7

```

```
Y11 es: 1.1042489459625537
```

Construya una funcion que retorne la varianza de un conjunto de datos ($n \geq 2$) cuando se va incorporando dato a dato. Use como primer par de datos los que se generan de la distribucion normal con $\mu = 3$ y $s = 0,3$. Use la misma semilla y los mismos parámetros de la distribucion para incorporar cada dato. Haga el proceso 50 veces y luego grafique la varianza como funcion de n .

```

1 import math
2 import numpy
3 import random
4 import matplotlib.pyplot as plt
5 def variance(n):
6     var2 = numpy.random.normal(3,0.3,2)
7     list_var = [var2[0], var2[1]]
8     list_n = numpy.random.normal(3,0.3,n)
9     for i in range(n):
10         list_var.append(list_n[i])
11         print(numpy.var(list_var))

```

```
-- Plotting the variance --  
12 import matplotlib.pyplot as plt  
13 plt.plot(list_var, range(n+2), 'pr')  
14 plt.xlabel('Varianza')  
15 plt.ylabel('N')  
16 plt.show()  
17 variance(50)
```

La cantidad de radiación absorbida puede medirse por la transmitancia. La relación entre transmitancia (T) y absorbancia (A) está dada por la ley Beer- Lambert (ley de Beer). Cree una función para la expresión que aparece en

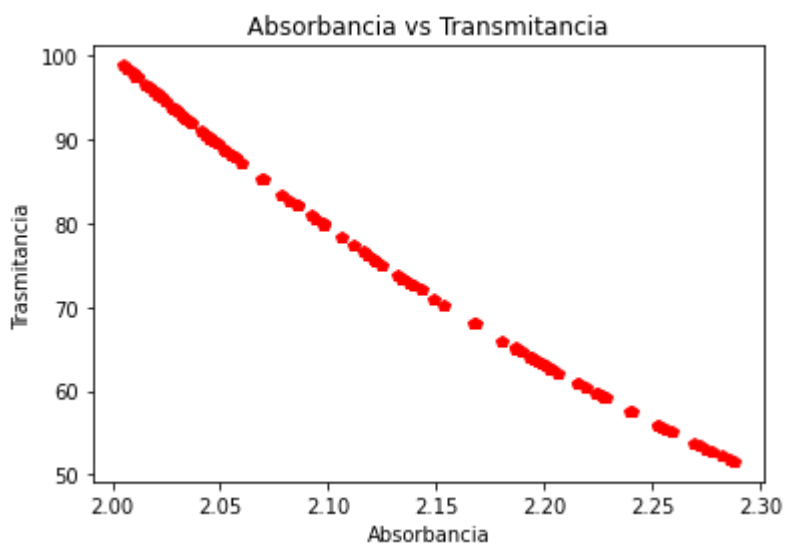
<https://www.electrical4u.com/what-is-transmittance/> y genere unos datos de T para determinar A. Grafique la función. Finalmente lea la sección titulada: ★★Why is Absorbance the Preferred Unit Over Transmittance?★★

```

1 import numpy
2 import matplotlib.pyplot as plt
3 import math
4 def absorbance(T, *args, **kwargs):
5     if str(type(T))[8:12] != 'list':
6         return f'ingresar lista de transmitancias'
7     else:
8         transmittance = []
9         absorbance = []
10        for i in range(len(T)):
11            transmittance.append(T[i])
12            A = 4 - math.log10(T[i])
13            absorbance.append(A)
14        plt.plot(absorbance, transmittance, 'pr')
15        plt.title('Absorbancia vs Transmitancia')
16        plt.xlabel('Absorbancia')
17        plt.ylabel('Trasmitancia')
18        plt.show()
19 import numpy
20 T = numpy.random.uniform(50,100,100)
21 absorbance(T =list(T))

```





✓ 0 s completado a las 22:03

