

PRACTICA FINAL 2da UNIDAD

1.- Ejercicio SYMPY

Calcule la integral definida de $f(x)=x^2 + 5$ en el rango de $[2,4]$. Calcule la expresión simbólica de dicha expresión.

```
1 from sympy import init_session
2 init_session(use_latex=True)
3 from sympy import (symbols,integrate)
4 x=symbols("x")
5 a=x**2+5
6 b=integrate(a,(x,2,4))
7 print(b)
```

These commands were executed:

```
>>> from __future__ import division
>>> from sympy import *
>>> x, y, z, t = symbols('x y z t')
>>> k, m, n = symbols('k m n', integer=True)
>>> f, g, h = symbols('f g h', cls=Function)
>>> init_printing()
```

Documentation can be found at <http://docs.sympy.org/1.3/>

86/3

2.- Ejercicio NUMPY

Cree un array de 5 filas por 4 columnas de valores aleatorios de 0 a 100, luego calcule:

- La suma de todos los elementos
- El valor mínimo dentro del arreglo
- La media del arreglo
- La mediana del arreglo
- La desviación estándar del arreglo

```
1 import numpy as np
2 a=np.random.random((5,4))
3 print(a)
4 suma=a.sum()
5 print("la suma es : ",suma)
6 minimo=a.min()
7 print("el valor minimo es: ",minimo)
8 media=np.mean(a)
9 print("la media es : ",media)
10 desviacion=np.std(a)
11 print("la desviación es : ",desviacion)
```

[[0.60462627 0.54785085 0.79486188 0.91256228]
[0.63083026 0.17034501 0.90062548 0.73865335]
[0.46228765 0.68829982 0.23632046 0.21657051]
[0.68682765 0.38190644 0.72572506 0.78279375]
[0.08467115 0.58621745 0.12652362 0.79464412]]

la suma es : 11.07314305048054
el valor minimo es: 0.08467114534535136
la media es : 0.553657152524027
la desviacion es : 0.25775910726619083

3.- Ejercicio MATPLOTLIB

Graficar la función seno y coseno en la misma grafica.

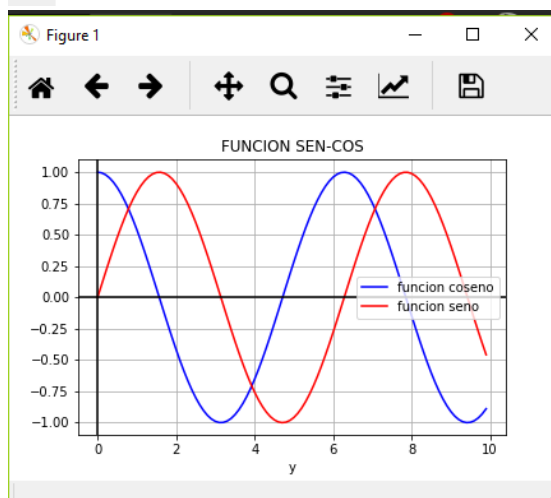
- Donde el eje x y el eje y sean los normales es decir $\sin(3.14) = 0$.
- Dicha grafica debe contener su respectiva leyenda.
- Nombrar a su grafica función sen-cos.

- El dominio debe ser $[0,10]$.
- Se pide que los ejes Esten marcador de color negro.

```

1 import matplotlib.pyplot as plt
2 import numpy as np
3
4 x=np.arange(0,10,0.1)
5 plt.title('FUNCION SEN-COS')
6 y=np.cos(x)
7 plt.plot(x,y,label="funcion coseno",color='b')
8 y=np.sin(x)
9 plt.plot(x,y,label="funcion seno",color='r')
10 plt.xlabel('x')
11 plt.ylabel('y')
12 plt.legend()
13 plt.grid()
14
15 plt.axhline(0,color="k")
16 plt.axvline(0,color="k")
17 plt.show()

```



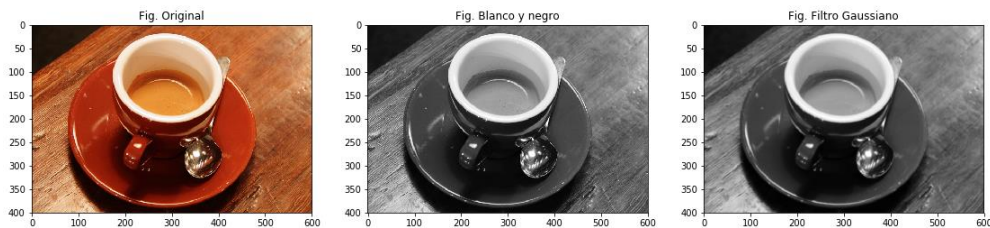
4.- Ejercicio de SKIMAGE

Importar una imagen del modulo skimage.data, convertirla a escala de grises y luego aplicarle un filtro gaussiano de skimage.filters.gaussian(imagen, sigma=1). Luego plotee la imagen original, la imagen en escala de grises y la imagen gaussiana en un solo gráfico. Agregue título a las gráficas.

```

1 from skimage import data, filters
2 from skimage.color import rgb2grey
3 import matplotlib.pyplot as plt
4
5 image=data.coffee()
6 imagegrey= rgb2grey(image)
7 imagegauss= filters.gaussian(imagegrey,sigma=1)
8
9 plt.subplot(1,3,1)
10 plt.imshow(image)
11 plt.title('Fig. Original')
12 plt.subplot(1,3,2)
13 plt.imshow(imagegrey,cmap='gray')
14 plt.title('Fig. Blanco y negro')
15 plt.subplot(1,3,3)
16 plt.imshow(imagegauss,cmap='gray')
17 plt.title('Fig. Filtro Gaussiano')
18 plt.show

```



5.- Ejercicio de PANDAS

Nombre	Nota 1	Nota 2	Nota 3
María	10	10	10
Juan	9	9	9
Rosa	8	8	8
Luis	7	7	7
Pedro	6	6	6
Ana	5	5	5
Eva	2	2	2

Se muestran las notas de un grupo de alumnos. Elaborar un algoritmo basado en Pandas, que, al ingresar el nombre de un alumno, te muestre su promedio final e indique si está aprobado ($N \geq 5$).

```

1 import pandas as pd
2 dict={"NOTA1":[10,9,8,7,6,5,2],"NOTA2":[10,9,8,7,6,5,2],"NOTA3":[10,9,8,7,6,5,2]}
3 notas=pd.DataFrame(dict)
4 notas.index=["MARIA","JUAN","ROSA","LUIS","PEDRO","ANA","EVA"]
5 print(notas)
6 promedio=0
7 nombre=input("INGRESE EL NOMBRE DE LA PERSONA QUE DESEA SABER SU PROMEDIO: ")
8 for i in range(3):
9     promedio=int(notas.ix[nombre,i])+promedio
10
11 print("El alumno "+nombre+" tiene de promedio: "+str(promedio/3))
12
13 if (promedio/3)>=5:
14     print(nombre+"ESTÁ APROBADO")
15 else:
16     print(nombre+"ESTÁ DESAPROBADO")
17 print(" ")

```

	NOTA1	NOTA2	NOTA3
MARIA	10	10	10
JUAN	9	9	9
ROSA	8	8	8
LUIS	7	7	7
PEDRO	6	6	6
ANA	5	5	5
EVA	2	2	2

INGRESE EL NOMBRE DE LA PERSONA QUE DESEA SABER SU PROMEDIO:
LUIS

C:/Users/Usuario/Downloads/Sin título5.py:9:

DeprecationWarning:

.ix is deprecated. Please use
.loc for label based indexing or
.iloc for positional indexing

See the documentation here:

<http://pandas.pydata.org/pandas-docs/stable/indexing.html#ix-indexer-is-deprecated>

promedio=int(notas.ix[nombre,i])+promedio

El alumno LUIS tiene de promedio: 7.0

LUIS ESTÁ APROBADO

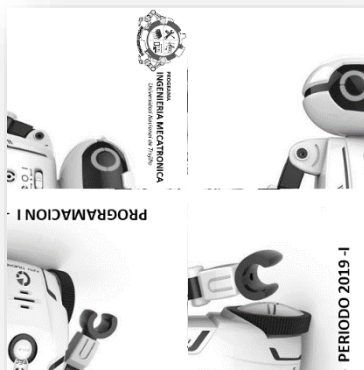
6.- Ejercicio de POO

Desarrollar un programa que conste de una clase padre Cuenta y dos subclases PlazoFijo y CajaAhorro. Definir los atributos titular y cantidad y un método para imprimir los datos en la clase Cuenta. La clase CajaAhorro tendrá un método para heredar los datos y uno para mostrar la información.

La clase PlazoFijo tendrá dos atributos propios, plazo e interés. Tendrá un método para obtener el importe del interés ($\text{cantidad} \times \text{interés} / 100$) y otro método para mostrar la información, datos del titular plazo, interés y total de interés.

Ejercicios Propuesto Por El Docente

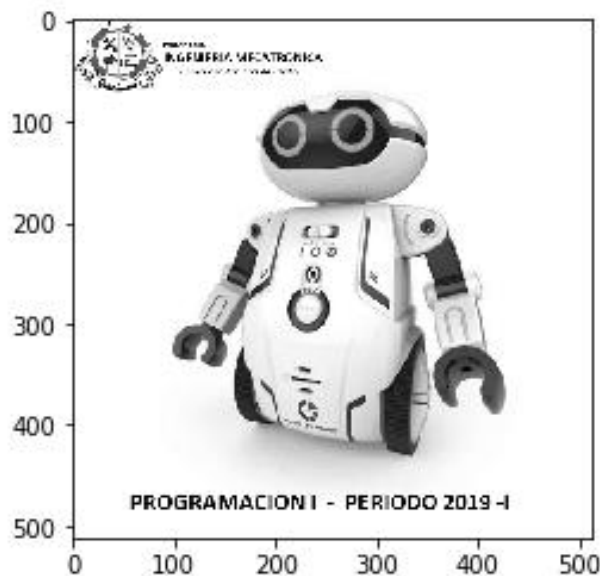
A la matriz de la imagen proporcionada de 512x512px, se pide hacer modificaciones en los cuadrantes para recrear la imagen original. Solo se permite hacer operaciones de corte (slicing) y transpuesta de Numpy. Muestre sus resultados utilizando matplotlib.



```

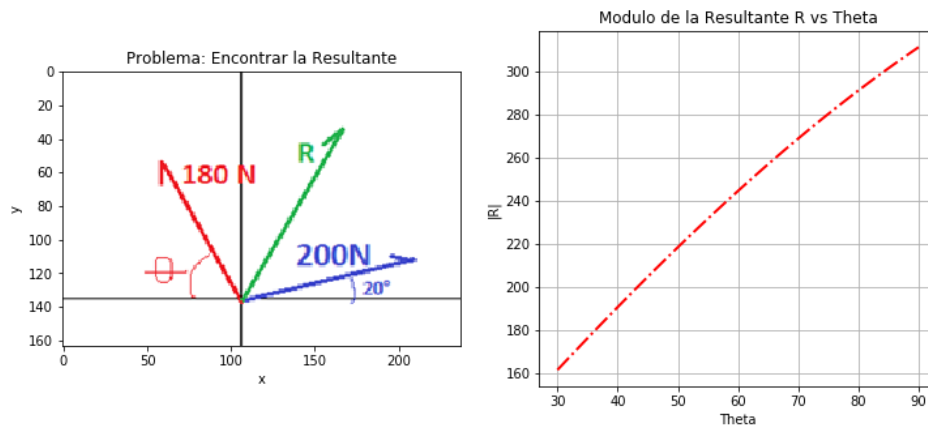
sinutil02.py sinutil03.py EJERCICIO 6.py 3
1 import matplotlib.pyplot as plt
2 import numpy as np
3 import skimage.io as io
4 im=io.imread("desarmado.png",as_gray="true")
5 plt.imshow(im,cmap="gray")
6 img_1=(im[0:255,0:255])
7 img_2=(im[0:255,256:512])
8 img_3=(im[256:512,0:255])
9 img_4=(im[256:512,256:512])
10 a=img_1.T
11 b=a[ : :-1]
12 c=img_2[ : :-1]
13 d=c.T
14 e=d[ : :-1]
15 f=e.T
16 g=f[ : :-1]
17 h=img_3[ : :-1]
18 i=h.T
19 j=i[ : :-1]
20 k=j.T
21 l=k[ : :-1]
22 m=l[ : :-1]
23 n=img_4[ : :-1]
24 v=n.T
25 plt.subplot(221)
26 plt.imshow(b,cmap="gray")
27 plt.subplot(222)
28 plt.imshow(g,cmap="gray")
29 plt.subplot(223)
30 plt.imshow(m,cmap="gray")
31 plt.subplot(224)
32 plt.imshow(v,cmap="gray")
33 plt.show()
34 img_1=b
35 img_2=g
36 img_3=m
37 img_4=v
38 im[0:255,0:255]=img_1
39 im[0:255,256:512]=img_2
40 im[256:512,0:255]=img_3
41 im[256:512,256:512]=img_4
42
43 plt.imshow(im,cmap="gray")
44 plt.show()

```



En el siguiente sistema de vectores se necesita saber el módulo de la resultante R en función del ángulo θ , el cual varía entre 30 a 90° . Realice la solución utilizando Python, muestre la grafica R vs θ usando matplotlib.

```
>>> import math
>>> math.pi
3.141592653589793
```



```
1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 x=np.arange(30,90,10)
5 a=np.pi
6 b=np.cos((160-x)*a/180)
7 y=np.sqrt(72400+72000*b)
8
9 plt.plot(x,y,'b,-.')
10 plt.title('Módulo de la resultante R vs Theta')
11 plt.xlabel('Theta')
12 plt.ylabel('Resultante')
13 plt.grid()
14 plt.show
```

