	Carátula para entrega de prácticas
Facultad de Ingeniería	Laboratorio de docencia

Laboratorios de computación salas A y B

Profesor: Adrian Ulises Mercado Martínez

Asignatura: Estructura de Datos y Algoritmos I

Grupo: 13

No de Práctica(s).

10

Integrante(s):

**Mondragón Carrillo Luis
Emir**

*No. de Equipo de
cómputo empleado:*

*No. de Equipo de
cómputo empleado:*

No. de Lista o Brigada:

Semestre:

2020-2

Fecha de entrega:

07/06/20

Observaciones:

CALIFICACIÓN: _____

INTRODUCCIÓN:

Para esta practica seguimos con la instrucción del lenguaje de programación de Python y en esta ocasión se nos muestran las bibliotecas, como graficar y las estructuras de iteración.

DESARROLLO:

1. Creamos un un archivo “py” donde utilizaremos la biblioteca math para poder hacer uso de funciones matematicas ademas damos un nombre a las funciones para después llamarlas y apoyarnos en las funcion de help para saber que hace cada función.

```
bilblioteca.py
1  #import math
2  #from math import *
3  from math import cos, pi
4  #x = math.cos(math.pi)
5  x = cos(pi)
6  print(x)
```

```
bilblioteca2.py
1  #import math
2
3  #print(dir(math))
4  #help(math.log)
5
6  import math as ma
7
8  x = ma.cos(ma.pi)
```

2. La primera estructura de integración en ser analizada es for, así como ver sus aplicaciones (arreglos, diccionarios, etc) además se implementará un else en

```
1  '''
2  For para listas
3  '''
4  def forlist():
5      for x in [1, 2, 3, 4, 5]:
6          print(x)
7
8      for x in ["uno", "dos", "tres", "cuatro", "cinco"]:
9          print(x)
10
11  '''
12  For para rangos
13  '''
14  def forrange():
15      for x in range(5):
16          print(x)
17
18      for y in range(-3,3):
19          print(y)
20
21      for z in range(-4, 2, 2):
22          print(z)
23
24      for i in range(5, 0, -1):
25          print(i)
26
27  un for
```

```

...
For para diccionarios
...
def fordic():
    diccionario = {'manzana':1, 'pera':3, 'uva':10}
    for clave, valor in diccionario.items():
        print(clave, " = ", valor)

    for clave in diccionario.keys():
        print(clave)

    for valor in diccionario.values():
        print(valor)

    for idx, x in enumerate(diccionario):
        print("El indice {} del elemneto {}".format(idx,x))

```

```

...
Else de For
...

def elsefor():
    for x in range(5):
        print(x)
    else:
        print("La cuenta se termino")

def elsefor2():
    for x in range(5):
        print(x)
        if x == 2:
            break
    else:
        print("La cuenta se termino")

if __name__ == "__main__":
    forlist()
    forrange()
    fordic()
    elsefor()
    elsefor2()

```

3. Estructura if

```
if.py
1  def numeroMayor(a, b, c):
2      if a > b and a > c:
3          print("El numero mayor es {}".format(a))
4      elif(b > c and b > a):
5          print("El nuemro mayor es {}".formatb(b))
6      else:
7          print("El numero mayor es {}".format(c))
8
9
10 if __name__ == "__main__":
11     a = int(input())
12     b = int(input())
13     c = int(input())
14     numeroMayor(a,b,c)
```

4. Estructura while

```
while.py
1  def factorial(n):
2      i = 2
3      temp = 1
4      while i <=n:
5          temp = temp * i
6          i = i + 1
7      return temp
8
9  if __name__ == "__main__":
10     a = int(input("Ingresa un numero: "))
11     print(factorial(a))
```

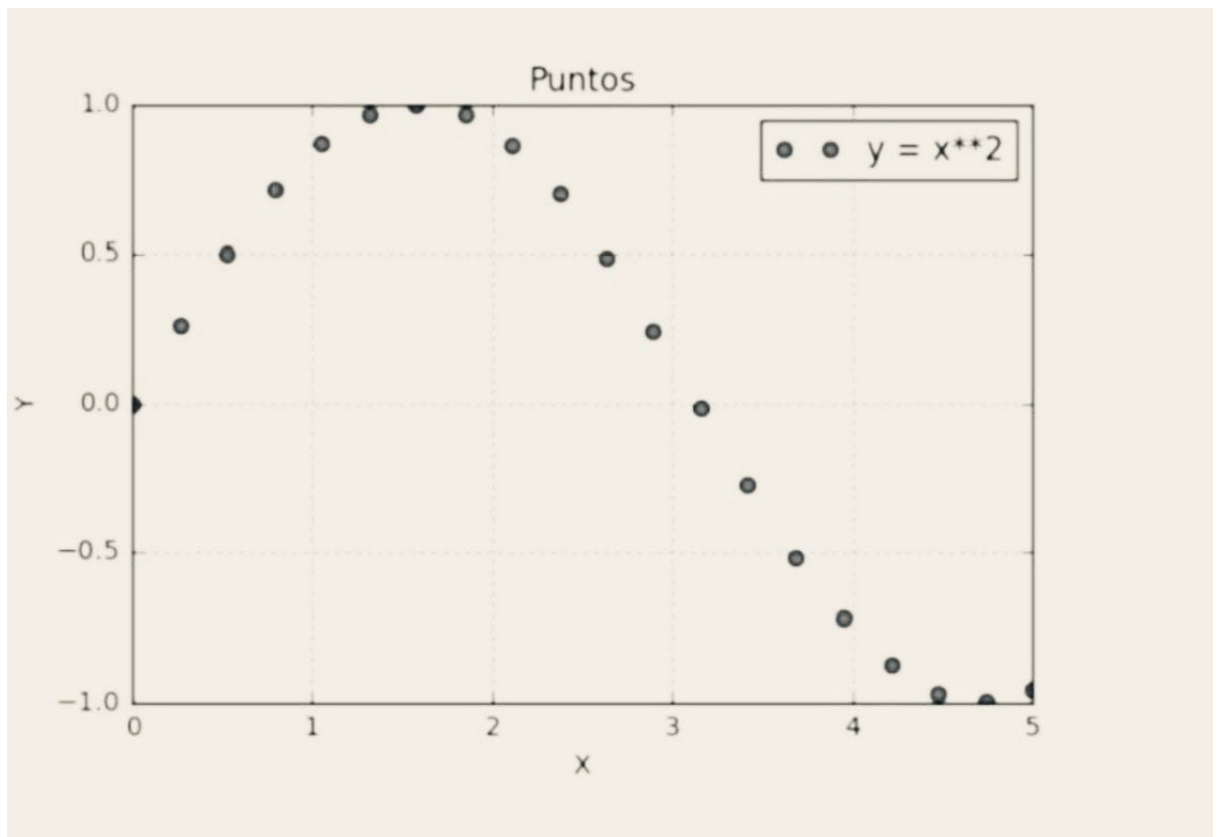
5. Para el caso de la pila vemos cómo aplicarlo en distintos lenguajes de programación donde las funciones es énviales están definidas en Python

cuando en c no pasa así.

```
1  def insertar(lista, dato):
2      lista.append(dato)
3
4
5  def borrar(lista):
6      dato = lista.pop()
7      return dato
8
9  def imprimir_pila(lista):
10     lista.reverse()
11     for x in lista:
12         print(x)
13     print()
14     lista.reverse()
15
16
17 def main():
18     pila = []
19     insertar(pila, "lista1")
20     insertar(pila, 2)
21     imprimir_pila(pila)
22     print()
23     print(borrar(pila))
24     imprimir_pila(pila)
25
26
27 if __name__ == "__main__":
28     main()
```

6. Usando la biblioteca matplotlib hacemos la gráfica.

```
1  import matplotlib.pyplot as plt
2  from mpl_toolkits.mplot3d import Axes3D
3  from math import sin
4  import numpy as np
5
6
7  x = 5
8
9
10 fig, ax = plt.subplots(facecolor='w', edgecolor='k')
11 ax.plot(x, sin(x), marker="o", color="r", linestyle='None')
12
13 ax.grid(True)
14 ax.set_xlabel('X')
15 ax.set_ylabel('Y')
16 ax.grid(True)
17 ax.legend(["y =x**2"])
18
19 plt.title('Puntos')
20 plt.show()
21
22 fig.savefig("Grafica.png")
```



CONCLUSIÓN:

En mi experiencia propia jamás había utilizado este lenguaje de programación, escuchaba que mucha gente le gustaba y les parecía mejor que otros, en algunos casos es más sencillo que C por sus funciones ya incorporadas pero es distinto a la hora de escribir el código por lo que es necesario adaptarse al mismo.