



## Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

# Laboratorios de computación salas A y B

*Profesor:* Ing. Adrián Ulises Mercado Martínez

*Asignatura:* Estructura de Datos y Algoritmos I

*Grupo:* 13

*No de Práctica(s):* 10

*Integrante(s):* Mejia Valdiviezo Eduardo Javier

*No. de Equipo de  
cómputo empleado:*

*No. de Lista o Brigada:* No. de lista: 30  
Brigada: 11

*Semestre:* 2020-2

*Fecha de entrega:* 07 / 06 / 2020

*Observaciones:*

**CALIFICACIÓN:** \_\_\_\_\_

## Práctica 10: Introducción a Python (II)

### Introducción

En esta práctica se revisan algunos conceptos básicos del lenguaje de programación Python: estructuras de control selectivas, estructuras de control repetitivas, bibliotecas y graficación.

### Objetivos

Aplicar las bases del lenguaje de programación Python.

### Desarrollo

#### Estructuras de control selectivas

```
1  def numeroMayor(a,b,c):
2      if a>b and a>c:
3          print("El número mayor es: {}".format(a))
4      elif(b>c and b>a):
5          print("El número mayor es: {}".format(b))
6      else:
7          print("El número 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)
```

En este programa sirve para encontrar el número mayor de tres números, los cuales son pedidos al usuario a través de la línea de comandos. Para implementar el algoritmo se hace uso de la sentencia if-elif-else.

#### Estructuras de control repetitivas

While:

```

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

```

En este programa se hace uso de la sentencia while para obtener el factorial de un número n.

For para listas:

```

1  """
2  For listas
3  """
4
5  def forList():
6      for x in [1,2,3,4,5]:
7          print(x)
8      for x in ["uno","dos","tres", "cuatro", "cinco"]:
9          print(x)
10

```

En Python, la sentencia for se usa mayormente para recorrer listas.

For para rangos:

```

11  """
12  For rangos
13  """
14
15  def forrange():
16      for x in range(5):
17          print(x)
18      for y in range(-3,3):
19          print(y)
20      for z in range(-4, 2, 2):
21          print(z)
22      for i in range(5,0,-1):
23          print(i)
24

```

For para diccionarios:

```

25  """
26  For diccionarios
27  """
28
29  def fordic():
30      diccionario = {'camisa': 1, 'pantalon' : 3, 'tenis': 10}
31      for clave, valor in diccionario.items():
32          print(clave,"=",valor)
33
34      for clave in diccionario.keys():
35          print(clave)
36
37      for valor in diccionario.values():
38          print(valor)
39
40      for idx,x in enumerate(diccionario):
41          print("El indice {} del elemento {}".format(idx,x))

```

La sentencia for también puede utilizarse para recorrer diccionarios. Recorriendo todos los elementos del diccionario dándole los valores de value y key a valor y clave respectivamente.

For-else:

```

44
45  def elsefor():
46      for x in range(5):
47          print(x)
48      else:
49          print("Se acabo")
50
51  def elsefor2():
52      for x in range(5):
53          print(x)
54          if x==2:
55              break
56      else:
57          print("La cuenta termino")

```

La sentencia else en el for se ejecuta después de que termina la iteración.

## Bibliotecas

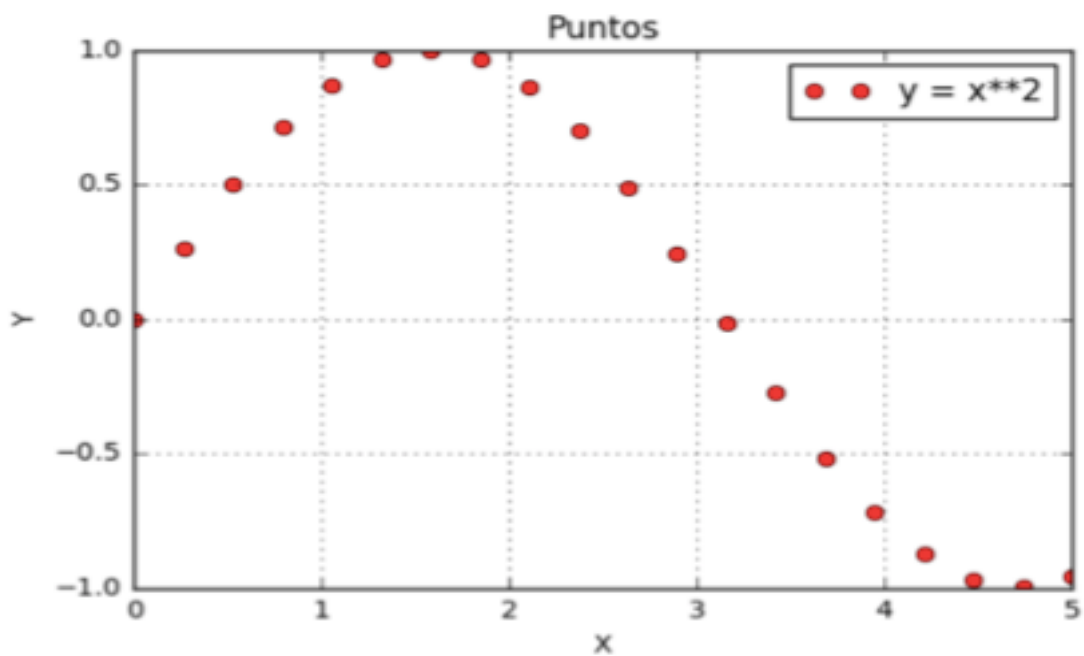
```

1  import math
2
3  print(dir(math))
4  help(math.log)
5
6  x = math.cos(math.pi)
7
8
9  from math import *
10 x = cos(pi)
11 print(x)
12
13 import math as ma
14 x = ma.cos(ma.pi)

```

Por medio de la función import se puede importar cualquier biblioteca de Python.

### Graficación



Para esta gráfica se usó la biblioteca Matplotlib que sirve para hacer gráficas en 2D y 3D. En este ejemplo se hace la grafica de la función seno con algunos generados entre 0 y 5.

## Conclusiones

Una práctica medianamente sencilla, conforme se va avanzando en los conocimientos de python se logra comprender un poco más la funcionalidad y se logra ver con mayor facilidad cómo usar ciertas funciones, muy parecido a otros lenguajes de programación.