

Laboratorio#3- Bryan Angulo López

OBJETIVO GENERAL DEL LABORATORIO:

INSTRUCCIONES PARA DESARROLLAR EL LABORATORIO:

El manejo de colección de datos en memoria es una característica importante de entender en todo lenguaje de programación, en Python se cuentan con varias estructuras para dicho objetivo, la más utilizada la estructura tipo <list>, con respecto a esta, realizar los siguientes puntos

Punto #1 (teórico) – [2.5%]

1. ¿De qué forma podemos agregar, eliminar, modificar y recorrer una estructura tipo list?

Para agregar un nuevo dato o elemento a una lista se utiliza el método “**append()**”, pero para añadir varios elementos debemos utilizar el método “**extend()**”.

Además, también es posible añadir un elemento en una posición concreta de una lista con el método “**insert (aquí ingresamos el índice, aquí ingresamos el elemento a incluir)**”, para los elementos en los cuales su índice sea mayor al índice ingresado se desplazaran una posición a la derecha.

Ejemplo:

```
>>> vocales = ['a', 'e', 'u']
```

```
>>> vocales.insert(2, 'i')
```

```
>>> vocales
```

```
['a', 'e', 'i', 'u']
```

Es posible modificar un elemento de una lista en Python con el operador de asignación “**=**”. Para ello, lo único que necesitas conocer es el índice del elemento que quieres modificar o el rango de índices:

Ejemplo:

```
>>> vocales = ['o', 'o', 'o', 'o', 'u']
```

```
# Actualiza el elemento del índice 0
```

```
>>> vocales[0] = 'a'
```

```
>>> vocales
```

```
['a', 'o', 'o', 'o', 'u']  
# Actualiza los elementos entre las posiciones 1 y 2  
>>> vocales [1:3] = ['e', 'i']  
>>> vocales  
['a', 'e', 'i', 'o', 'u']
```

Para eliminar un elemento de una lista podemos utilizar la sentencia “**del**” a partir de su índice.

Ejemplo:

```
# Eliminar el elemento del índice 1  
>>> vocales = ['a', 'e', 'i', 'o', 'u']  
>>> del vocales [1]  
>>> vocales  
['a', 'i', 'o', 'u']  
# Elimina los elementos con índices 2 y 3  
>>> vocales = ['a', 'e', 'i', 'o', 'u']  
>>> del vocales[2:4]  
>>> vocales  
['a', 'e', 'u']  
# Elimina todos los elementos  
>>> del vocales[:]  
>>> vocales  
[]
```

Además de la sentencia **del**, podemos usar los métodos “**remove()**” el cual elimina la primera ocurrencia que se encuentre del elemento en una lista y “**pop([i])**” el cual obtiene el elemento cuyo índice sea igual a *i* y lo elimina de la lista. Si no se especifica ningún índice, recupera y elimina el último elemento.

Finalmente, es posible eliminar todos los elementos de una lista a través del método “**clear()**”:

```
>>> letras = ['a', 'b', 'c']  
>>> letras.clear()  
>>> letras  
[]
```

Recorrer una lista:

Una forma sencilla de recorrer una lista o cualquier objeto iterable en Python es usando el bucle “**for**”.

En caso de que queramos recorrer la lista desde y hacia algún índice específico, podemos hacerlo usando la función “**range()**”.

La función **range(start, stop, step)** devuelve una secuencia que comienza desde el valor de start, termina en el valor de stop y da el paso igual al argumento step cuyo valor predeterminado es 1.

Ejemplo:

```
mylist = ['a','b','c','d','e','f','g']  
for x in range(2,len(mylist)-1):  
    print(mylist[x])
```

Producción:

```
c  
d  
e  
f
```

2. De qué forma podemos contar la cantidad de elementos que se encuentran en una colección tipo <list>

Para contar la cantidad de elementos de una lista podemos utilizar la función incorporada **“len()”** en Python, la cual nos devuelve el número total de elementos en una lista, sin tener en cuenta el tipo de elementos que contiene.

También podemos usar la función len() para contar el número de elementos de los otros tres tipos de datos integrados que ofrece Python, a saber, Tuple, Set y Dictionary.

3. Que entiende por indexación de los elementos en una colección de datos.

Una indexación es una forma de encontrar el índice o posición de la primera aparición de algún elemento dentro de una lista, para esto se utiliza el método **“index”**, cabe recalcar que si el elemento se encuentra repetido esto no nos va a indicar los índices de las repeticiones, sino, que solo nos mostrara el índice de la primera aparición.

Punto #2 – [2.5%]

1. Elabore un programa tipo consola que permita ejemplificar las acciones de agregar, eliminar, modificar y recorrer una estructura tipo <list>, el contexto del desarrollo queda a criterio del estudiante.

```
Lista = ["a","c","r","p","z","l","f","o","s"]

print("La lista inicial es: ")
print(Lista)

Lista.append("adios")
Lista.insert(5,"m")
print("Lista con datos agregados: ")
print(Lista)

Lista[2:5] = ["cambie", "Yo tambien","X3"]
print("Lista modificada: ")
print(Lista)

del Lista[4:6]
print("Lista sin los datos eliminados: ")
print(Lista)

print("Se recorre la lista en:")
for x in range(4,len(Lista)-2):
    print(Lista[x])

print("El tamaño de la lista es:")
print(len(Lista))
```

Resultado:

```
II/Laboratorio3.py"
La lista inicial es:
['a', 'c', 'r', 'p', 'z', 'l', 'f', 'o', 's']
Lista con datos agregados:
['a', 'c', 'r', 'p', 'z', 'm', 'l', 'f', 'o', 's', 'adios']
Lista modificada:
['a', 'c', 'cambie', 'Yo tambien', 'X3', 'm', 'l', 'f', 'o', 's', 'adios']
Lista sin los datos eliminados:
['a', 'c', 'cambie', 'Yo tambien', 'l', 'f', 'o', 's', 'adios']
Se recorre la lista en:
l
f
o
El tamaño de la lista es:
9
PS C:\Users\casa> []
```

Código en visual Studio Code:

```
Laboratorio3.py •
C: > Users > casa > OneDrive - Universidad Internacional de las Américas (UIA) > 1-Universidad > U > I\

1  Lista=["a","c","r","p","z","l","f","o","s"]
2  print("La lista inicial es: ")
3  print(Lista)
4
5  Lista.append("adios")
6  Lista.insert(5,"m")
7  print("Lista con datos agregados: ")
8  print(Lista)
9
10 Lista[2:5] = ["cambie", "Yo tambien","X3"]
11 print("Lista modificada: ")
12 print(Lista)
13
14 del Lista[4:6]
15 print("Lista sin los datos eliminados: ")
16 print(Lista)
17
18 print("Se recorre la lista en:")
19 for x in range(4,len(Lista)-2):
20     print(Lista[x])
21
22 print("El tamaño de la lista es:")
23 print(len(Lista))
24

PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN  TERMINAL

La lista inicial es:
['a', 'c', 'r', 'p', 'z', 'l', 'f', 'o', 's']
Lista con datos agregados:
['a', 'c', 'r', 'p', 'z', 'm', 'l', 'f', 'o', 's', 'adios']
Lista modificada:
['a', 'c', 'cambie', 'Yo tambien', 'X3', 'm', 'l', 'f', 'o', 's', 'adios']
Lista sin los datos eliminados:
['a', 'c', 'cambie', 'Yo tambien', 'l', 'f', 'o', 's', 'adios']
Se recorre la lista en:
l
f
o
El tamaño de la lista es:
9
PS C:\Users\casa> 
```

RECURSOS PARA USAR EN EL LABORATORIO:

- Visual Studio Code

ENTREGABLES

- Documento PDF con las respuestas teóricas versionado en Laboratorio #3 de plataforma GitHub.
- Código fuente versionado en Laboratorio #3 de plataforma GitHub.