

**UNIVERSIDAD CENTRAL DE VENEZUELA**

**DEPARTAMENTO DE INVESTIGACIÓN**

**DE OPERACIONES Y COMPUTACIÓN**

**PROGRAMACIÓN (0790)**

**CARLOS GONZÁLEZ**

**APELLIDO Y NOMBRE:** SANOJA IRIHEY

**CEDULA DE IDENTIDAD:** 27.600.072

**INFORME**

**MÉTODO DE ORDENAMIENTO DE LA BURBUJA (BUBBLESORT)**

En Python, el ordenamiento de burbuja es un algoritmo que te permite ordenar valores de un arreglo. Funciona revisando cada elemento con su adyacente. Si ambos elementos no están ordenados, se procede a intercambiarlos, si por el contrario los elementos ya estaban ordenados se dejan tal como estaban.

En código que se muestra en el enlace es el siguiente:

from random import sample

# Importamos un Método de la biblioteca random para generar listas aleatorias

lista = list(range (100)) # Creamos la lista base con números del 1 al 100

# Creamos una lista aleatoria con sample

# (8 elementos aleatorios de la lista base)

vectorbs = sample(lista,8)

def bubblesort(vectorbs):

"""Esta función ordenara el vector que le pases como argumento con el Método de Bubble Sort"""

# Imprimimos la lista obtenida al principio (Desordenada)

print("El vector a ordenar es:",vectorbs)

n = 0 # Establecemos un contador del largo del vector

for \_ in vectorbs:

n += 1 #Contamos la cantidad de caracteres dentro del vector

for i in range(n-1):

# Le damos un rango n para que complete el proceso.

for j in range (0, n-i-1):

# Revisa la matriz de 0 hasta n-i-1

if vectorbs[j] > vectorbs[j+1]:

vectorbs[j], vectorbs[j+1] = vectorbs[j+1], vectorbs[j]

# Se intercambian si el elemento encontrado es mayor

# Luego pasa al siguiente

print ("El vector ordenado es: ", vectorbs)

bubblesort(vectorbs)

## El vector a ordenar es: [53, 1, 38, 93, 74, 83, 4, 18]

## El vector ordenado es: [1, 4, 18, 38, 53, 74, 83, 93]

El código es un ejemplo de cómo se puede ordenar una lista de números utilizando el algoritmo de ordenamiento de burbuja en Python.

Primero se importa el método (**sample)** de la biblioteca (**random**) para generar una lista aleatoria de 8 elementos a partir de una lista base que contiene números del 1 al 100. Luego se define la función (**bubblesort**) que recibe como argumento el vector a ordenar y utiliza el algoritmo de ordenamiento de burbuja para ordenar los elementos del dicho vector.

La función (**bubblesort)** recibe como parámetro el vector a ordenar y utiliza dos ciclos FOR anidados para comparar cada elemento con su adyacente y realizar el intercambio si es necesario. El primer ciclo (**for**) se encarga de contar el número de elementos en el vector y el segundo ciclo (**for**) se encarga de realizar las comparaciones y los intercambios necesarios, imprimiendo el vector desordenado al principio y luego imprime el vector ordenado.

El algoritmo de ordenamiento de burbuja es un algoritmo simple pero ineficiente para ordenar listas. Funciona comparando pares adyacentes de elementos y si están en el orden incorrecto, los intercambia. Este proceso se repite hasta que la lista esté completamente ordenada.