

Ejercicio 1:

Suponga que usted tiene que ordenar la siguiente lista de números: [15, 5, 4, 18, 12, 19, 14, 10, 8, 20] ¿Cuál de las siguientes listas representa la lista parcialmente ordenada tras tres pasadas completas del ordenamiento por inserción?

- A. [4, 5, 12, 15, 14, 10, 8, 18, 19, 20]
- B. [15, 5, 4, 10, 12, 8, 14, 18, 19, 20]
- C. [4, 5, 15, 18, 12, 19, 14, 10, 8, 20]
- D. [15, 5, 4, 18, 12, 19, 14, 8, 10, 20]

Ejercicio 2:

Suponga que usted tiene que ordenar la siguiente lista de números: [11, 7, 12, 14, 19, 1, 6, 18, 8, 20] ¿Cuál de las siguientes listas representa la lista parcialmente ordenada tras tres pasadas completas del ordenamiento por selección?

- A. [7, 11, 12, 1, 6, 14, 8, 18, 19, 20]
- B. [7, 11, 12, 14, 19, 1, 6, 18, 8, 20]
- C. [11, 7, 12, 14, 1, 6, 8, 18, 19, 20]
- D. [11, 7, 12, 14, 8, 1, 6, 18, 19, 20]

Ejercicio 3:

Dada la siguiente lista de números [14, 17, 13, 15, 19, 10, 3, 16, 9, 12] ¿Cuál respuesta corresponde al contenido de la lista después de la segunda partición de acuerdo al algoritmo de ordenamiento rápido?

- A. [9, 3, 10, 13, 12]
- B. [9, 3, 10, 13, 12, 14]
- C. [9, 3, 10, 13, 12, 14, 17, 16, 15, 19]
- D. [9, 3, 10, 13, 12, 14, 19, 16, 15, 17]

Ejercicio 4

Dada la siguiente implementación de inserción, modificarla para que ordene de mayor a menor

```
def ordenamientoPorInsercion(unalista):  
    for indice in range(1,len(unalista)):  
  
        valorActual = unalista[indice]  
        posicion = indice  
  
        while posicion>0 and unalista[posicion-1]<valorActual:  
            unalista[posicion]=unalista[posicion-1]  
            posicion = posicion-1  
  
        unalista[posicion]=valorActual  
  
unalista = [54,26,93,17,77,31,44,55,20]  
ordenamientoPorInsercion(unalista)  
print(unalista)
```

