

entrega-ejercicio-04

November 4, 2024

1 Entrega 4

- **Ejercicio:** implementar el algoritmo de búsqueda binaria utilizando un proceso iterativo y explicar su funcionamiento

La función `binarySearch` busca un elemento en una lista ordenada dividiéndola repetidamente en mitades.

Comienza calculando el punto medio y comparando su valor con el elemento buscado. Si el valor en el medio es igual al elemento, lo encuentra y devuelve su índice; si es menor, ignora la mitad izquierda y sigue en la derecha; si es mayor, ignora la derecha y continúa en la izquierda. Este proceso se repite hasta encontrar el elemento o hasta que no queden elementos por buscar.

- **Indicación:** podéis utilizar este código sugerido y comentar los pasos del algoritmo

```
[ ]: # Búsqueda binaria iterativa
# Devuelve la posición dónde se encuentra el elemento, -1 si no existe

def binarySearch(arr, l, r, x):
    # Mientras el índice izquierdo (l) sea menor o igual que el derecho (r)
    while l <= r:
        # Calculamos el índice del punto medio
        mid = l + (r - l) // 2

        # Verificamos si x coincide con el punto medio
        if arr[mid] == x:
            return mid

        # Si x es mayor, podemos ignorar el segmento izquierdo
        elif arr[mid] < x:
            l = mid + 1

        # Si x es menor, entonces ignoramos el segmento derecho
        else:
            r = mid - 1

    # Si no hemos encontrado x, devolvemos -1
    return -1

# Ejemplo
# Lista de elementos ordenados
```

```
arr = [2, 3, 4, 10, 40]
# Elemento que queremos encontrar
x = 10

# Llamada a la función binarySearch
result = binarySearch(arr, 0, len(arr)-1, x)

# Imprimimos el resultado
if result != -1:
    print("Elemento encontrado en la posición % d" % result)
else:
    print("Elemento no encontrado")
```

Elemento encontrado en la posición 3