

# Laboratorio # 1

Juan Diego Sique Martínez

Julio 2018

## 1. Búsqueda lineal

Input: arreglo «array» y caracter a buscar «operador»

```
busqueda(array, operador) {  
    posicion = null  
    for(i = 1, i = len(array), i++) {  
        if(array[i] = operador) {  
            posicion = i  
            break  
        }  
    }  
    return posicion  
}
```

La INVARIANTE DE CICLO consiste en que el valor de la variable **i** nos indica la posición o lugar en el arreglo donde no se encuentra nuestro caracter deseado, y al momento de encontrarla rompe el ciclo retornándonos dicho valor **i**.

## 2. Multiplicación de matrices

Input: matriz A y matriz B Output: matriz C

COMPLEJIDAD	INSTRUCCIÓN DE CÓDIGO
n	For i from 1 to n:
n × p	For j from 1 to p:
n × p	Let sum = 0
n × p	For k from 1 to m:
n × p × m	Set sum <- sum + A[i][k] * B[k][j]
n × p	Set C[i][j] <- sum
1	return C

Como pudimos apreciar el tiempo de  $O(n \times p \times m)$  ya que se recorren tres ciclos que corresponden a las dos dimensiones de las matrices y la tercera a una suma acumulativa para poder obtener el resultado de la multiplicación. Si

las tres matrices poseen el mismo número de elementos podríamos decir que la multiplicación consecutiva de  $n$  por sí misma tres veces da como resultado  $O(n^3)$ .

### 3. Ordenación por «Bubble-sort»

---

**Algorithm 1** Bubble sort algorithm

---

```
S is an array of integer
for  $i$  in  $1 : \text{length}(S) - 1$  do
  for  $j$  in  $(i + 1) : \text{length}(S)$  do
    if  $S[i] > S[j]$  then
      swap  $S[i]$  and  $S[j]$ 
    end if
  end for
end for
```

---

#### 3.1. ¿Cuál es el «worst-case running time» de este algoritmo?

Es apreciable el ciclo anidado donde el primer **for** y sus instrucciones incluidas se ejecutan al menos  $n - 1$  veces. La complejidad del algoritmo crece cuando el segundo **for** se ejecuta también  $n - 1$  veces resultando  $(n - 1)^2$ . Por lo tanto en el peor de los casos tendrá una complejidad del tipo  $O(n^2)$ .

#### 3.2. ¿Cómo se compara a el «running time» de «insertion sort»?

Se compara en la presencia de un ciclo anidado que al ejecutarse  $n - 1$  de manera consecutiva producen una complejidad de tipo  $O(n^2)$ .