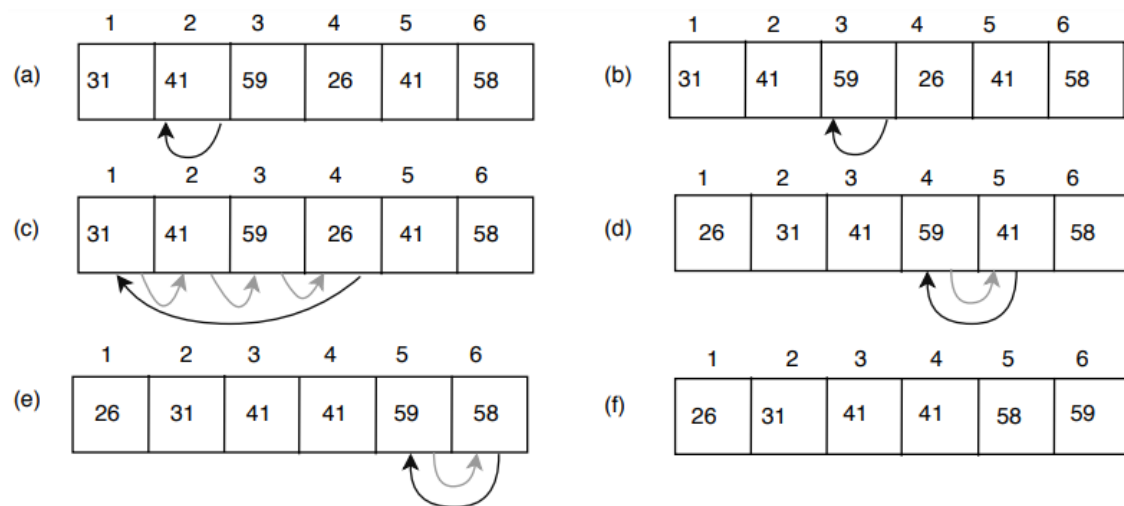


Algoritmos 01

September 2018

1 Tarea 01 Algoritmos

1) Using Figure 2.2 as a model, illustrate the operation of INSERTION-SORT on the array $A = \langle 31, 41, 59, 26, 41, 58 \rangle$.



2) Rewrite the INSERTION-SORT procedure to sort into nonincreasing instead of nondecreasing order.

```
def insertionSortInverse(array):  
    for i in range(1, len(array)):  
        aux = array[i]  
        j = i - 1  
        while j > -1 and aux > array[j]:  
            array[j + 1] = array[j]  
            j = j - 1  
        array[j + 1] = aux
```

3) Consider the *searching problem*:

Input: A sequence of n numbers $A = \langle a_1, a_2, \dots, a_n \rangle$ and a value v .

Output: An index i such that $v = A[i]$ or the special value NIL if v does not appear in A .

Write pseudocode for *linear search*, which scans through the sequence, looking for v . Using a loop invariant, prove that your algorithm is correct. Make sure that your loop invariant fulfills the three necessary properties.

Programa: Búsqueda secuencial

Variables

T=10:entero

```

A[T]:arreglo de tamaño t
temp,i,j,n:entero
x:binario
Inicio
  //Llenar arreglo
  para i=0 hasta i< T incremento 1 hacer
    A[i]=númeroaleatorio
  fin para
  //Busqueda lineal
  leer n
  x=falso
  para i=0 hasta i< T incremento 1 hacer
    si A[i] = n entonces
      escribir "Valor encontrado"
      escribir "Posición:■, i
      x=verdadero
    fin si
  fin para
  si x=falso entonces
    escribir ■No se encontró el número■
  fin si
Fin

```

Inicializacion: Se inicia llenando un arreglo de tamaño $T=10$ con valores aleatorios luego se lee el valor a buscar y se compara este valor con el valor en la posicion inicial del arreglo.

Mantenimiento: Se compara el valor obtenido al inicio con el siguiente término del arreglo $A[i+1]$, repitiendo uno a uno el proceso de manera secuencial, si se encuentra que los terminos son iguales se cambia el valor de x de falso a verdadero y se escribe "valor encontrado" junto con su posicion en el arreglo y el algoritmo continúa.

Terminacion:

El algoritmo finaliza cuando ha recorrido cada uno de los elementos del arreglo $i = T+1$, si no encontro ninguna coincidencia entre el valor Inicial y los valores del arrelgo entonces $x = \text{falso}$ y escribe "No se encontro el número".

4) Consider the problem of adding two n -bit binary integers, stored in two n -element arrays A and B . The sum of the two integers should be stored in binary form in an $(n+1)$ -element array C . State the problem formally and write pseudocode for adding the two integers.

Declaration of A , B and C :

```

A[0] ... A[n-1] (length = n)
B[0] ... B[n-1] (length = n)
C[0] ... C[n] (length = n+1)
A[0] and B[0] are the most significant bits.

```

Pseudocode:

```

Carry = 0
For i = n - 1 to 0
  C[i+1] = (A[i] + B[i] + Carry) mod 2
  Carry = (A[i] + B[i] + Carry) / 2
C[0] = Carry

```