**A21 ArregloOrdenado**

**ANÁLISIS**

Comparar Si eTotal = MAX

Mostrar (“No hay espacio.”)

Si no

Comparar Si eTotal = 0 ó cNombre > aNombres [eTotal]

Incrementar eTotal + 1

Asignar aNOmbres[eTotal] = cNombre

Si no

Comparar Si BuscarBinario (aNombres, eTotal, cNombre) <> 0

Mostrar (“Ya existe”, eNombre)

Si no

Inicializar eRecorre = eTotal

Repetir

Asignar aNombres[eRecorre + 1] = aNombres[eRecorre]

Decrementar eRecorre – 1

Mientras eRecorre > 1 y aNombres[eRecorre] > cNombre

Asignar aNombres[eRecorre + 1] = cNombre

Incrementar eTotal + 1

**PSEUDOCÓDIGO**

Const

MAX = 10

Tipo

TaNombres = arreglo [1…MAX] de cadena

Var

cNombre, cNuevo: cadena

eTotal, eOpcion: entero

aNombres: TaNombres

**Procedimiento InsertarOrdenada(ES aNombres: TaNombres, ES eTotal: entero, E cNombre: cadena)**

Var

eRecorre: entero

Inicio

Si (eTotal = MAX) entonces

Escribir (“No hay espacio”)

De lo contrario

Si (eTotal = 0 o cNombre > aNombres [eTotal]) entonces

Hacer eTotal = eTotal + 1

Hacer aNombres[eTotal] = cNombre

De lo contrario

Si BuscarBinario (aNombres, eTotal, cNombre) <> 0 entonces

Escribir(“Ya existe”, cNombre)

De lo contrario

Hacer eRecorre = eTotal

Mientras (eRecorre >= 1 y aNombres[eRecorre] > cNombre) hacer  
 Hacer aNombre[eRecorre + 1] = aNombres[eRecorre]

Hacer eRecorre = eRecorre + 1

Fin Mientras

Hacer aNombres[eRecorre + 1] = cNombre

Hacer eTotal = eTotal + 1

Fin Si

Fin Si

Fin Si

Fin // Procedimiento InsertarOrdenada

**Procedimiento BorrarOrdenada(ES aNombres: TaNombres, ES eTotal: entero, E cNombre: cadena)**

Var

ePos, eRecorre: entero

Inicio

Si (eTotal = 0) entonces

Escribir (“El arreglo está vacío.”)

De lo contrario

Hacer ePos = BuscarBinario (aNombres, eTotal, cNombre)

Si (ePos = 0) entonces

Escribir (“No existe.”)

De lo contrario

Desde eRecorre = ePos hasta eTotal – 1 hacer  
 Hacer aNombres[eRecorre + 1] = aNombres[eRecorre + 1]

Hacer eRecorre = eRecorre + 1

Fin Desde

Fin Si

Fin Si

Fin // Procedimiento BorrarOrdenada

**Procedimiento ModificarOrdenada(ES aNombres: TaNombres, E eTotal: entero, E cNombre: cadena, E cNuevo: cadena)**

Inicio

Si (eTotal = 0) entonces

Escribir (“El arreglo está vacío.”)

De lo contrario

Si (BuscarBinario (aNombres, eTotal, cNombre) = 0) entonces

Escribir (“No existe”, cNombre, “dentro del arreglo.”)

De lo contrario

Si (BuscarBinario (aNombres, eTotal, cNuevo) <> 0) entonces

Escribir (“Ya existe”, cNombre, “dentro del arreglo.”)

De lo contrario

Ejecutar (BorrarOrdenada (aNombres, eTotal, cNombre)

Ejecutar (InsertarOrdenada (aNombres, eTotal, cNuevo)

Fin Si

Fin Si

Fin Si

Fin // Procedimiento ModificarOrdenada

**Función BuscarBinario (E aNombres: TaNombres, E eTotal: entero, E cNombre: cadena): entero**

Var

eMenor, eMayor, eMedio: entero

Inicio

Si ( eTotal = 0 ) entonces

Regresar 0

De lo contrario

Hacer eMenor = 1

Hacer eMayor = eTotal

Hacer eMedio = eMayor + eMenor) DIV 2

Mientras (eMenor <= eMayor y aNombres[eMedio] <> cNombre )

Si (cNombre > aNombres[eMedio]) entonces

Hacer eMenor = eMedio + 1

De lo contrario

Hacer eMayor = eMedio - 1

Fin Si

HacereMedio = ( eMenor + eMayor) DIV 2

Fin Mientras

Si ( aNombres [eMedio] = cNombre ) entonces

Regresar eMedio

De lo contrario

Regresar 0

Fin Si  
Fin Si

Fin Función// BuscarBinario