

### Ejercicio práctico Ordenamiento y búsqueda en vectores.

1. Realizar un algoritmo que lea y almacene en memoria por medio de vectores los datos de cinco estudiantes; documento, nombre y edad, se debe realizar un procedimiento que permita buscar un estudiante por medio de su respectivo documento de identidad, si el elemento a buscar se encuentra en el vector que almacena el documento del estudiante, imprimir un mensaje que indique que se ha encontrado el documento y la posición en la que se encuentra en dicho vector, así mismo mostrar el resto de datos del estudiante, de lo contrario imprimir un mensaje que indique que el documento buscado no se encuentra en el vector.

```
1  Proceso Busqueda_Estudiante
2
3      // Se definen y declaran variables a utilizar
4      Definir auxNom como Texto
5      Definir i, j, auxDoc, auxEdad, numDoc, mitad como Entero
6
7      // Se dimensionan los vectores
8      Dimension vecDocEst[5];
9      Dimension vecNomEst[5];
10     Dimension vecEdaEst[5];
11
12     // Se inicializan variables
13     inicio ← 1;
14     final ← 5;
15     encontro ← Falso;
16
17
18     // Se repite el procedimiento para los 5 estudiantes
19     Para i ← 1 hasta 5 Hacer
20         limpiarpantalla
21
22         Escribir " Ingreso de datos del estudiante # ", i;
23         Escribir " ";
24
25         Escribir " Digite el documento : "; Sin saltar Leer vecDocEst[i];
26         Escribir " Digite el nombre : "; Sin saltar Leer vecNomEst[i];
27         Escribir " Digite la edad : "; Sin saltar Leer vecEdaEst[i];
28
29     FinPara
```

## Vectores – Algoritmos de Programación

---

```
30
31 // Se ordena el vector vecDocEst[] con el metodo Burbuja
32 Para i ← 1 hasta 4 Hacer
33     Para j ← (i + 1) hasta 5 Hacer
34         // Se realiza el procedimiento de ordenamiento - intercambio del metodo Burbuja
35         Si (vecDocEst[i] > vecDocEst[j]) Entonces
36             auxDoc ← vecDocEst[i];
37             vecDocEst[i] ← vecDocEst[j];
38             vecDocEst[j] ← auxDoc;
39             auxNom ← vecNomEst[i];
40             vecNomEst[i] ← vecNomEst[j];
41             vecNomEst[j] ← auxNom;
42             auxEda ← vecEdaEst[i];
43             vecEdaEst[i] ← vecEdaEst[j];
44             vecEdaEst[j] ← auxEda;
45         FinSi
46     FinPara
47 FinPara
48 limpiarpantalla
49
50
51 // Se imprimen los datos ordenados de los 5 estudiantes ingresados en los vectores; vecDocEst[], vecNomEst[], vecEdaEst[]
52 Para i ← 1 hasta 5 Hacer
53     Escribir " Imprimir datos del estudiante # ", i;
54     Escribir " ";
55     Escribir " Documento : ", vecDocEst[i];
56     Escribir " Nombre : ", vecNomEst[i];
57     Escribir " Edad : ", vecEdaEst[i];
58 FinPara
59
60
61 // Procedimiento binario para buscar un dato por medio del vector - vecDocEst[]
62 Escribir " ";
63 Escribir " ";
64 Escribir "Digite el documento del estudiante que desea buscar : "; sin saltar Leer numDoc; //Lee el documento a buscar en el vector vecDocEst[]
65 Escribir " ";
66
67 Mientras ((encontro = Falso) y (inicio ≤ final)) Hacer
68     mitad ← Trunc((inicio + final)/2)
69     Si numDoc = vecDocEst[Mitad] Entonces
70         encontro ← Verdadero;
71     Sino
72         Si numDoc < vecDocEst[Mitad] Entonces
73             final ← mitad - 1;
74         Sino
75             inicio ← mitad + 1;
76         Finsi
77     FinSi
78 FinMientras
```

## Vectores – Algoritmos de Programación

---

```
88
89 // Se imprimen los datos con la validación del metodo de busqueda binaria.
90 Si Encontro = Verdadero Entonces
91     Escribir "El Documento Nro. ", numDoc, " del estudiante, se encuentra en la posición ", mitad, " del vector";
92     Escribir " ";
93     Escribir "Los demas datos del estudiante son :";
94     Escribir " ";
95     Escribir "El Nombre del estudiante es :", vecNomEst[mitad];;
96     Escribir "La Edad del estudiante es :", vecEdaEst[mitad];
97
98 Sino
99     Escribir " ";
100     Escribir "El Documento Nro. ", numDoc, " no se encuentra en ninguna posición del vector ";
101 FinSi
102 Escribir " ";
103
104 FinProceso
```

---