

Practica de Vectores Nº 2

Desarrolle un programa que realice la búsqueda dentro de un vector de números reales de un valor dado por el usuario. En caso de conseguir el dato emitir un mensaje.

Pasos:

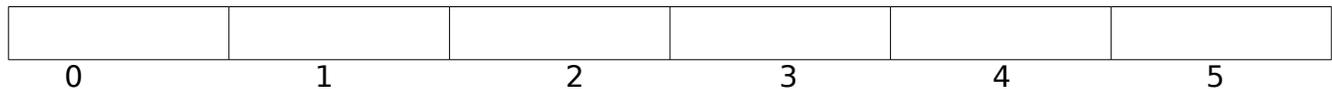
1. Leer el vector (datos)
2. Solicitar el dato a buscar
3. Buscar el elemento.

Solución por pasos

Paso 1: Leer el vector (datos).

Inicialmente debemos decidir la dimensión de nuestro vector, asumamos 6 posiciones (pueden utilizar las que consideren convenientes). Luego designamos un nombre o identificador valido para nuestra variable del tipo arreglo unidimensional (datos).

datos



Recordando que las posiciones del vector se manejan desde 0 hasta N-1, tendremos un vector datos cuyas posiciones varían entre 0 y 5, para un total de 6 elementos.

Si deseamos leer la posición 3 del vector datos:

```
scanf("%f",&datos[3]);
```

La misma instrucción se repite para la lectura de las distintas posiciones, por tanto si deseamos leer todas las posiciones (0-5) solo debemos utilizar una estructura de repetición cuyo contador varíe en este rango.

```
for (i=0;i<6;i++)
```

```
scanf("%f",&datos[i]);
```

lo cual es equivalente a definir una constante $N = 6$ y decir:

```
for (i=0;i<N;i++)
```

```
scanf("%f",&datos[i]);
```

El programa completo hasta este punto quedaria:

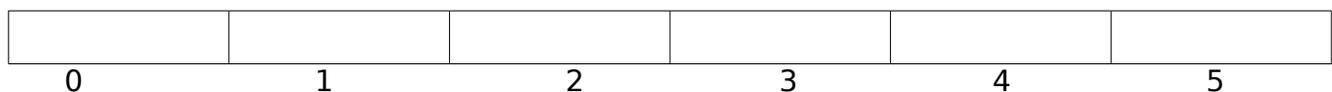
```
#include<stdio.h> // definición de librería
#define N 6 // utilizamos una constante para definir la longitud del vector
int main(){
    int i; // definimos el vector datos de N elementos y el contador i
    float datos[N];
    for (i=0;i<N;i++){
        printf("Inserte el valor de la posicion %d del vector datos: ",i);
        scanf("%f",&datos[i]);
    }
    return 0;
}
```

Paso 2: Solicitar el valor a buscar

Esto es equivalente a leer una variable (valor_busca) que debe ser de tipo real puesto que el vector es de números reales.

Paso 3: Buscar el elemento.

datos



Para realizar la búsqueda es necesario recorrer todos los elementos del vector y preguntar.

“Es la posición actual igual al valor buscado?”

De ser cierto, emitiremos un mensaje que indique haber conseguido el valor

De lo contrario, no emitiremos mensaje alguno.

```
for (i=0;i<N;i++) // recorreremos el vector desde la posicion 0 hasta la 5
    if (datos[i]==valor_busca) //“Es la posición actual igual al valor buscado?”
        printf("Encontrado\n"); // de ser cierto, mensaje de éxito.
```

El programa completo quedaria asi:

```
#include<stdio.h> // definición de librería
#define N 6 // utilizamos una constante para definir la longitud del vector
int main(){
    int i; // definimos el vector datos de N elementos y el contador i
    float datos[N],valor_busca;
    for (i=0;i<N;i++){
        printf("Inserte el valor de la posicion %d del vector datos: ",i);
        scanf("%f",&datos[i]);
    }
    printf("\nIndique el valor que desea buscar: ");
    scanf("%f",&valor_busca);
    for (i=0;i<N;i++) // recorremos el vector desde la posicion 0 hasta la 5
        if (datos[i]==valor_busca) //"Es la posición actual igual al valor buscado?"
            printf("Encontrado\n"); // de ser cierto, mensaje de éxito.
    return 0;
}
```

Este código tiene un inconveniente, solo emite mensaje en caso de conseguir el valor buscado, de lo contrario no emite mensaje alguno. Deberíamos ampliar nuestro programa para que en caso de llegar al final del programa y no conseguir el valor buscado, escriba por pantalla "valor no encontrado".

Lo podemos lograr utilizando una variable que indique si el valor ha sido conseguido o no (un centinela). Si conseguimos el valor, el centinela cambia de valor, de lo contrario se mantiene igual, así al finalizar la estructura de repetición solo preguntaremos: "Existió un cambio en el centinela? ". De ser cierto, quiere decir que el valor fue encontrado, de lo contrario el valor no fue encontrado.

```
centinela=0; //centinela=0 indica inicialmente que el valor buscado no está en el vector
for (i=0;i<N;i++) // recorremos el vector desde la posicion 0 hasta la 5
    if (datos[i]==valor_busca) //"Es la posición actual igual al valor buscado?"
        centinela=1; // cambiamos el centinela puesto que conseguimos el valor
```

```

if (centinela!=0) // preguntamos si el centinela ha cambiado
    printf("Valor encontrado"); // si ha cambiado quiere decir que se encontró
else
    printf("Valor no encontrado"); // si no ha cambiado --> no se encontró

```

El programa completo quedaria asi

```

#include<stdio.h> // definición de librería
#define N 6 // utilizamos una constante para definir la longitud del vector

int main(){
    int i,centinela;
    float datos[N],valor_busca;
    for (i=0;i<N;i++){
        printf("Inserte el valor de la posicion %d del vector datos: ",i);
        scanf("%f",&datos[i]);
    }
    printf("\nIndique el valor que desea buscar: ");
    scanf("%f",&valor_busca);
    centinela=0;
    for (i=0;i<N;i++) // recorremos el vector desde la posicion 0 hasta la 5
        if (datos[i]==valor_busca) //"Es la posición actual igual al valor buscado?"
            centinela=1;
    if (centinela!=0)
        printf("Valor encontrado");
    else
        printf("Valor no encontrado");
    return 0;
}

```

Ejercicios propuestos para el 30/06:

1. Modificar el código para que indique el número de veces que se consigue el valor buscado dentro del vector.
2. Modificar el código para que indique además la posición en que se encuentra el valor buscado.