

B. Acomoda el número

Puntos	8.8	Límite de memoria	256 MiB
Límite de tiempo (caso)	1s	Límite de tiempo (total)	1m0s
Entrada/Salida	Consola	Tamaño límite de entrada (bytes)	10 KiB

Tienes una lista desordenada con N números distintos los cuales tienen valores entre 0 y M . Llamemos al primer número de esta lista a_0 . Debes ordenar los números de la lista y ver en qué posición queda el número a_0 . En este problema la primera posición de la lista es la posición 0 y la última es la posición $N - 1$.

Problema

Escribe un programa que dada la lista de N números determine en qué posición queda el elemento a_0 (el primero de la lista) después que ésta se ordena.

Entrada

Tu programa debe leer del teclado la siguiente información:

- En la primera línea el número entero N , la cantidad de números en la lista.
- En la segunda línea los N números de la lista separados cada uno por un espacio.

Salida

Tu programa debe escribir en la pantalla un único número entero que representa la posición final del elemento a_0 en la lista ordenada.

Restricciones

$$1 < N \leq 2,000,000 \quad 0 \leq M \leq 10^9$$

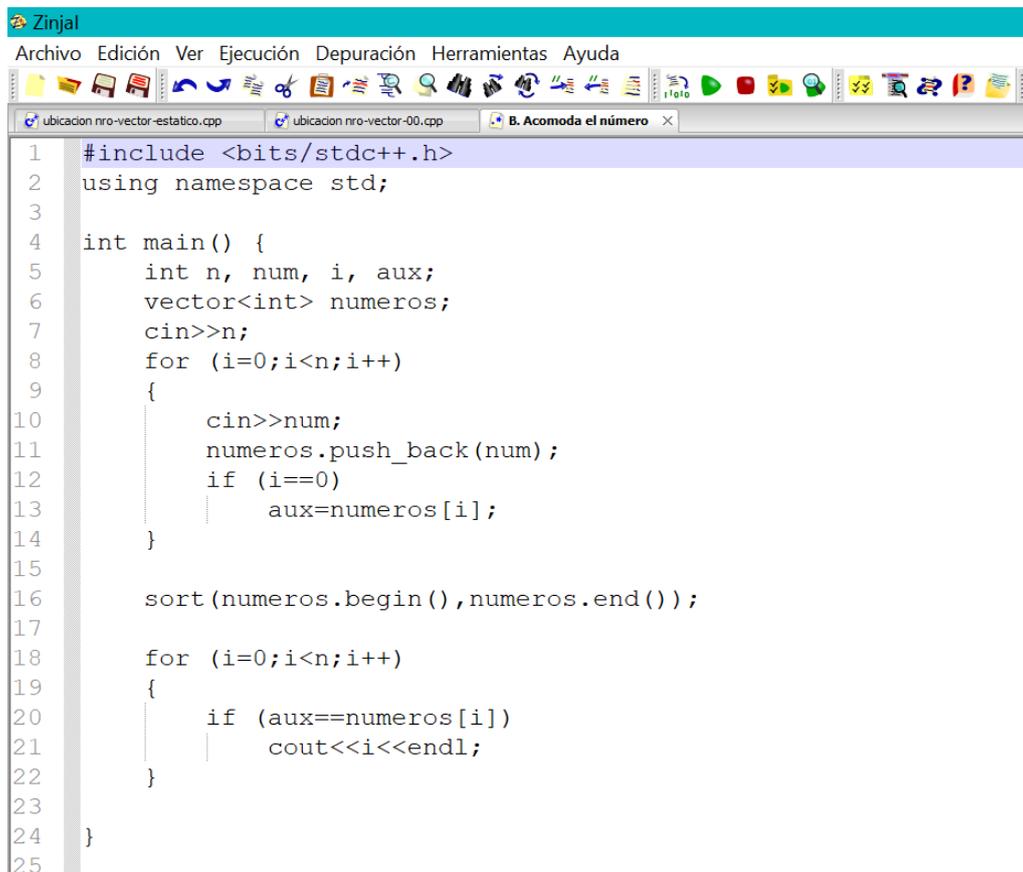
Ejemplo

Entrada	Salida	Descripción
5 14 2 1 17 23	2	Recuerda que las posiciones de la lista se cuentan a partir de 0 . La lista ordenada queda 1, 2, 14, 17, 23, de modo que 14 , que era el primer elemento en la lista desordenada, quedó en la posición 2 .

Evaluación

Para un grupo de casos con valor de **51** puntos $N \leq 5,000$.

Ejemplo Vector dinámico



```
1 #include <bits/stdc++.h>
2 using namespace std;
3
4 int main() {
5     int n, num, i, aux;
6     vector<int> numeros;
7     cin>>n;
8     for (i=0;i<n;i++)
9     {
10         cin>>num;
11         numeros.push_back(num);
12         if (i==0)
13             aux=numeros[i];
14     }
15
16     sort(numeros.begin(), numeros.end());
17
18     for (i=0;i<n;i++)
19     {
20         if (aux==numeros[i])
21             cout<<i<<endl;
22     }
23 }
24
25
```

Ejemplo Vector Estático

Zinjal

Archivo Edición Ver Ejecución Depuración Herramientas Ayuda



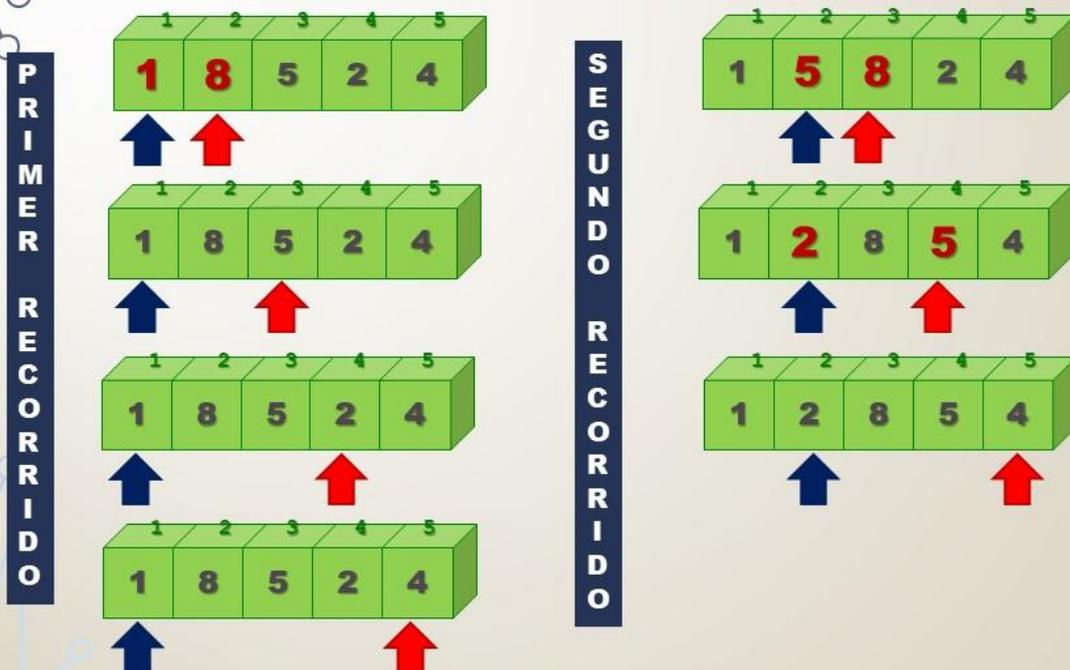
ubicacion nro-vector-estatico.cpp x ubicacion nro-vector-00.cpp* B. Acomoda el número

```
1  #include <bits/stdc++.h>
2  using namespace std;
3
4  int main() {
5      int n, num, i, aux, aux2, j;
6      int numeros [20];
7      cin>>n;
8      for (i=0;i<n;i++)
9          {
10             cin>>numeros[i];
11             if (i==0)
12                 aux=numeros[i];
13         }
14
15         for (i=0;i<n;i++)
16         {
17             cout << numeros[i]<< endl;
18         }
19         //ordenamiento por selección
20         for (i=0; i<n-1; i++)
21             for (j=i+1; j<n; j++)
22                 {
23                     if (numeros[i]>numeros[j])
24                         {
25                             aux2=numeros[i];
26                             numeros[i]=numeros[j];
27                             numeros[j]=aux2;
28                         }
29                 }
30         for (i=0;i<n;i++)
31         {
32             cout << numeros[i]<< endl;
33             if (aux==numeros[i])
34                 cout<<" posicion del a0  "<<i<<endl;
35         }
36     }
37 }
38
```

ORDENACIÓN. SELECCIÓN (1)

- El algoritmo de *selección* consiste en recorrer el arreglo (comparando cada posición con las siguientes) e intercambiar elementos cuando sea necesario.
 1. Se compara la primera posición del arreglo con las restantes y se intercambia su contenido cuando corresponda.
 2. Se compara la segunda posición del arreglo con las restantes y se intercambia su contenido cuando corresponda.
 3. Este proceso se repite para cada posición del arreglo.
 4. El algoritmo finaliza cuando se evalúa la penúltima posición.

ORDENACIÓN. SELECCIÓN (2)



ORDENACIÓN. SELECCIÓN (3)

TERCER RECORRIDO



CUARTO RECORRIDO

