

**Instrucciones:** lee detenidamente cada pregunta y responde de manera correcta; aplicando

tus conocimientos del capítulo 6 del libro de Deitel; correspondientes a las secciones 6.6 al 6.8.

¿Cuál de los algoritmos de ordenación tiene una complejidad igual a

**(n\*log2\*n)**?

5/5

QuickSort. Burbuja.

Inserción. N.d.a

Selección.



Si un algoritmo de ordenación tiene eﬁciencia logarítmica entonces este 5/5

algoritmo es que un algoritmo con eﬁciencia lineal.

Peor. Igual.

Mejor.



El algoritmo de ordenación que aplica la técnica **divide y vencerás**

corresponde a:

0/5

N.d.a. QuickSort. Selección. Inserción

Shell.

Respuesta correcta

QuickSort.



Para que el algoritmo de ordenación **QuickSort** tenga un alto desempeño 5/5

depende:

De la elección del Pivote. Del tamaño de la lista.

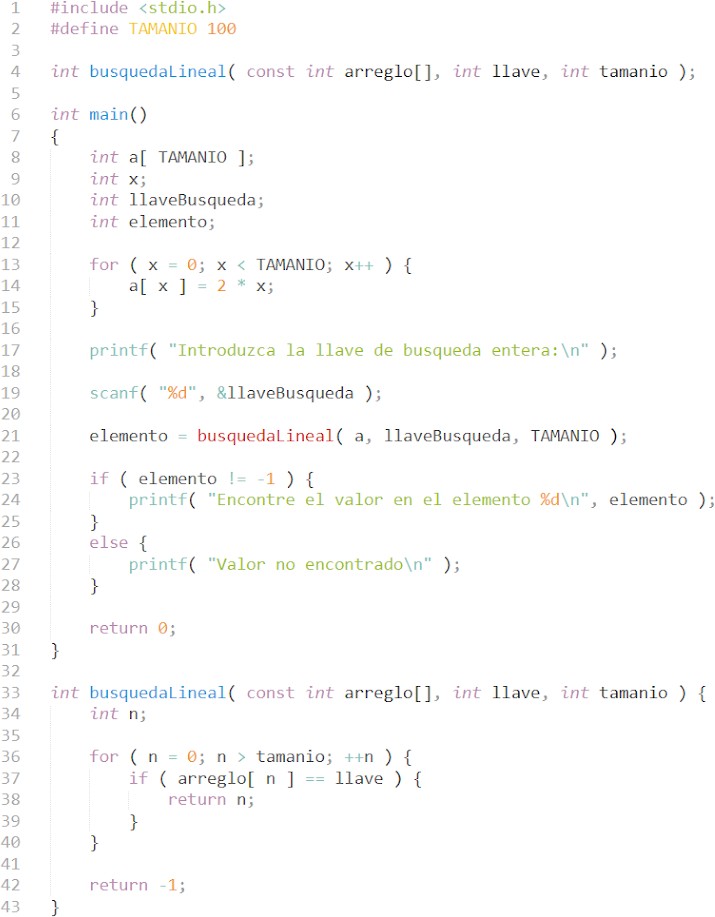
De la función del algoritmo sea recursivo. Que la lista se encuentre ordenada.

N.d.a.

# ¿En qué número de línea se encuentra el error? Para que el siguiente

programa cumpla correctamente con la búsqueda lineal en un arreglo.

5/5



36



Utilizando el algoritmo de **BÚSQUEDA BINARIA.** ¿Cuántas pasadas se

necesita para encontrar el valor **88**? Para la lista de abajo:

5/5

**[8, 13, 17, 26, 44, 56, 88, 97]**

4

3

1

2

0



¿Qué algoritmo de ordenación se está usando?

5/5

Si inicialmente, se tiene la siguiente lista:

**[47, 3, 21, 32, 56, 92]**

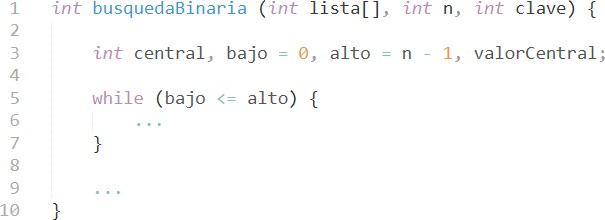
Después de dos pasadas de un algoritmo de ordenación, la lista se ha

quedado dispuesto así:

**[3, 21, 47, 32, 56, 92]**

QuickSort. Inserción. Selección. Burbuja.

N.d.a.



Cuando la variable ***bajo*** es mayor que la variable ***alto*** en

una **BÚSQUEDA BINARIA** signiﬁca que ya no existe vector a buscar la clave.

5/5

Falso.

Verdadero.



Este algoritmo en la primera pasada busca el menor elemento desde ***v[0]*** a 5/5 ***v[n - 1]*** y lo deposita en la posición ***v[0]***, luego se busca el menor elemento desde ***v[1]*** a ***v[n - 1]*** y lo deposita en la posición ***v[1]*** y, así sucesivamente

para las siguientes pasadas del algoritmo.

Inserción. QuickSort. Shell.

Burbuja.

Selección. N.d.a.



Si se tiene una lista de 5.000 elementos el número de elementos

examinados en el peor de los casos en una **búsqueda binaria** seria:

0/5

14

11

41

18

81

Respuesta correcta

14



La codiﬁcación de la **BÚSQUEDA BINARIA** sólo se puede realizar de forma 5/5

ITERATIVA.

Verdadero.

Falso.



Si los datos están almacenados en un archivo, el proceso de ordenación se5/5

llama ordenación: .

externa



En el método de hundimiento para ordenar arreglos, el ciclo interno cumple 5/5

la función para controlar el número de por pasada.

comparaciones



¿Cuál es el algoritmo de ordenación que mejora el método de **Inserción**

**Directa**?

5/5

Selección. Shell.

Insercción. QuickSort.

N.d.a.



Para que se aplique correctamente la **BÚSQUEDA BINARIA** la lista NO tiene5/5

que estar ordenada ya sea de forma ascendente o descendente.

Verdadero.

Falso.



El algoritmo que se basa en las lecturas sucesivas de la lista a ordenar, 5/5

comparando el elemento inferior de la lista con los restantes y efectuando

de posiciones cuando el orden resultante de la comparación no sea el correcto.

1. recorrido
2. inserción
3. intercambio
4. mezcla
5. selección



En terminología de ordenación el elemento por el cual está ordenado un 5/5

conjunto de datos (o se esta buscando) se denomina: .

clave



La eﬁciencia en una **BÚSQUEDA SECUENCIAL** es:

5/5

log2n (logaritmo de base 2 por n) N.d.a.

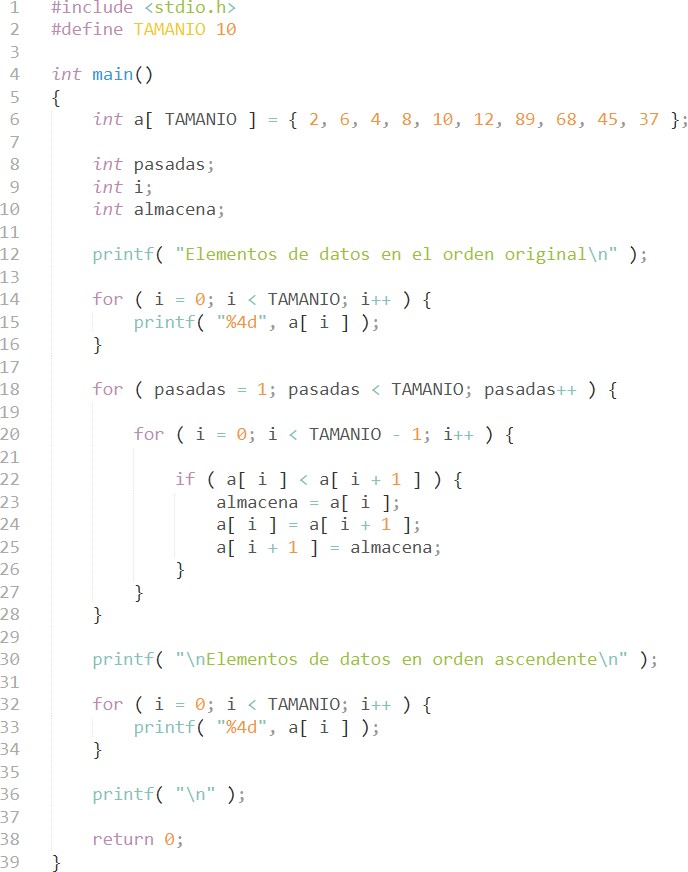
1

n

n2 (n al cuadrado)

# ¿Este programa ordena los valores de un arreglo en orden?

0/5



Ascendente. Descendente.

Tiene error de sintaxis. No ordena la lista.

Respuesta correcta



Descendente.

La búsqueda compara cada elemento de un arreglo con la *clave* 5/5

*de búsqueda*.

lineal

Este formulario se creó en Facultad Politecnica UNA.

[Formularios](https://www.google.com/forms/about/?utm_source=product&utm_medium=forms_logo&utm_campaign=forms)

