



**SEP**  
SECRETARÍA DE  
EDUCACIÓN PÚBLICA

**UTP**  
DIRECCIÓN GENERAL DE UNIVERSIDADES  
TECNOLÓGICAS y POLITÉCNICAS

**UTNG**  
Universidad Tecnológica  
del Norte de Guanajuato  
Organismo Público Descentralizado del Gobierno del Estado  
"Educación y progreso para la vida"

## Universidad Tecnológica del Norte de Guanajuato "UTNG"

Organismo público descentralizado del Gobierno del Estado de  
Guanajuato

"Educación y progreso para la vida"

### Docente:

Gabriel Barron Rodriguez

### Programa educativo:

Licenciatura en Ingeniería en Tecnologías de la Información e  
Innovación Digital

### Materia:

Estructura de datos

### Grupo:

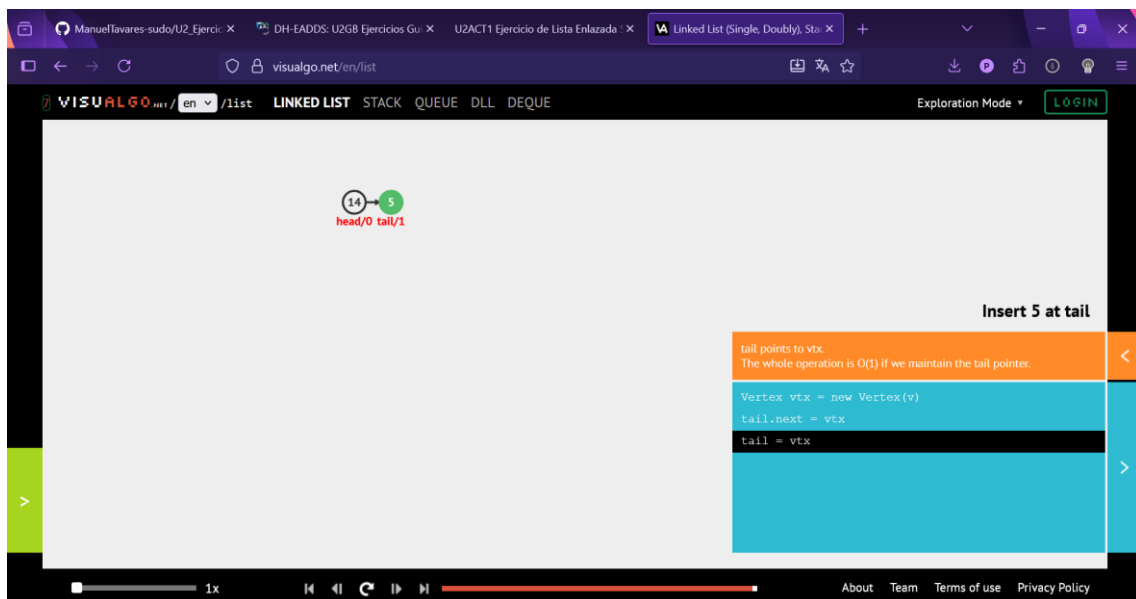
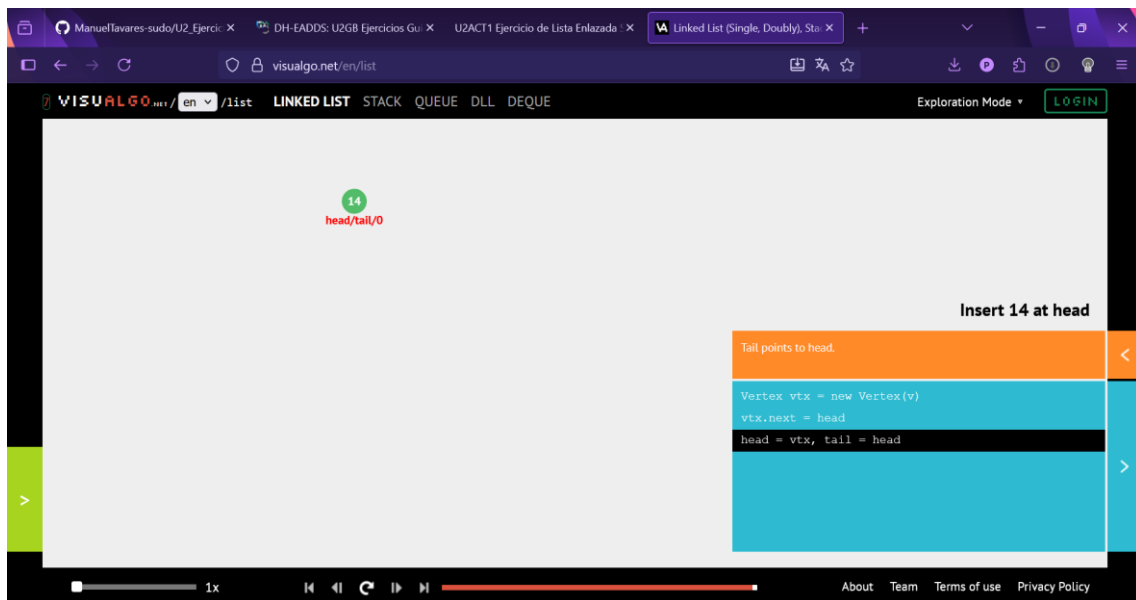
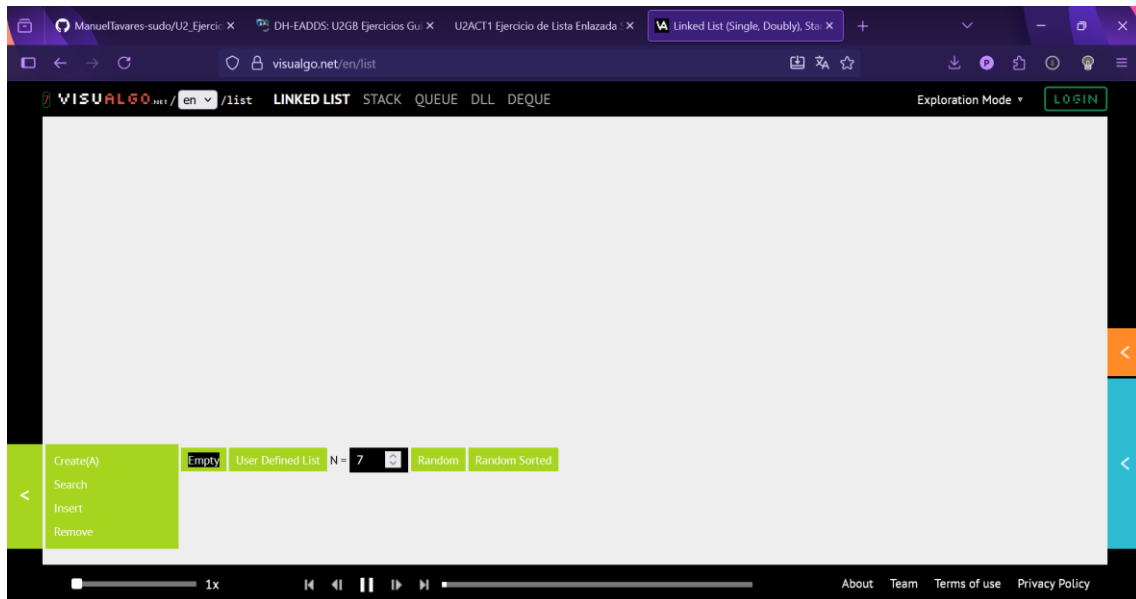
GTID141

### Alumno:

Manuel Alejandro Tavares Hernández

### Fecha:

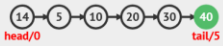
3 de octubre de 2025



ManuelTavares-sudo/U2\_Ejercic... DH-EADD5: U2GB Ejercicios Gu... U2ACT1 Ejercicio de Lista Enlazada... Linked List (Single, Doubly), Sta... +

visualgo.net/en/list

VISUALGO .net /en /list LINKED LIST STACK QUEUE DLL DEQUE Exploration Mode LOGIN



head/0 tail/5

### Insert 40 at tail

tail points to vtx.  
The whole operation is  $O(1)$  if we maintain the tail pointer.

```
Vertex vtx = new Vertex(v)
tail.next = vtx
tail = vtx
```


1x

About Team Terms of use Privacy Policy

ManuelTavares-sudo/U2\_Ejercic... DH-EADD5: U2GB Ejercicios Gu... U2ACT1 Ejercicio de Lista Enlazada... Linked List (Single, Doubly), Sta... +

visualgo.net/en/list

VISUALGO .net /en /list LINKED LIST STACK QUEUE DLL DEQUE Exploration Mode LOGIN



head/0 tail/4

### Remove i = N-1 (Tail)

Delete tmp (the previous tail) then update the tail pointer to pre (the current tail). The whole process is  $O(N)$  just to find the pre pointer.

```
if empty, do nothing
Vertex pre = head
tmp = head.next
while (tmp.next != null)
    pre = pre.next; tmp = tmp.next;
pre.next = null
delete tmp, tail = pre
```


1x

About Team Terms of use Privacy Policy

ManuelTavares-sudo/U2\_Ejercic... DH-EADD5: U2GB Ejercicios Gu... U2ACT1 Ejercicio de Lista Enlazada... Linked List (Single, Doubly), Sta... +

visualgo.net/en/list

VISUALGO .net /en /list LINKED LIST STACK QUEUE DLL DEQUE Exploration Mode LOGIN



head/0 tail/2

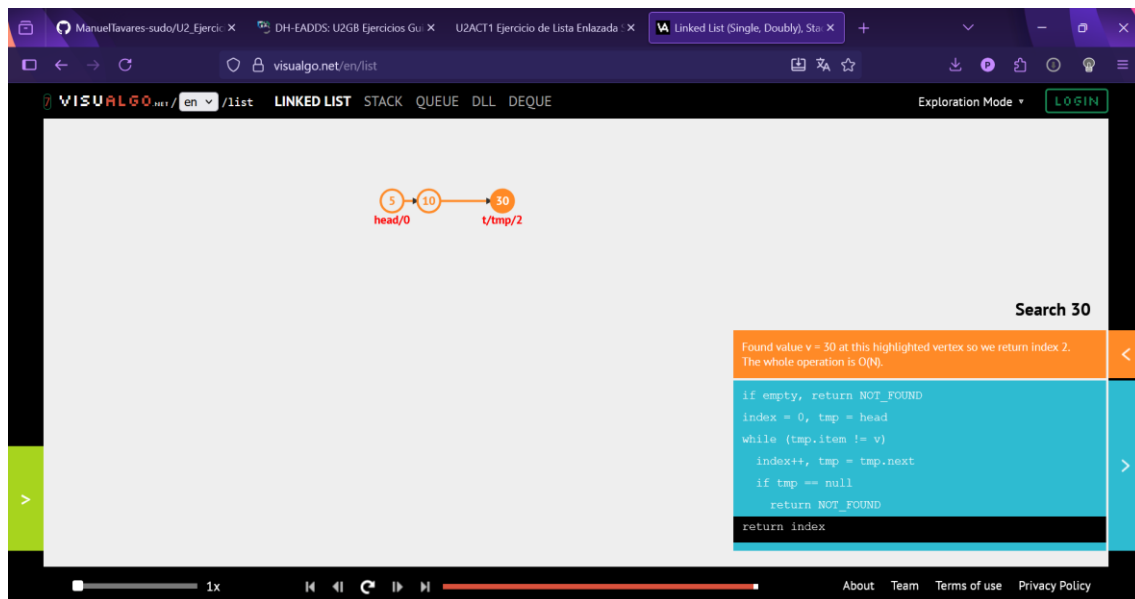
### Remove index 2

Re-layout the Linked List for visualization (not in the actual Linked List).  
The whole process is still  $O(N)$ .

```
if empty, do nothing
Vertex pre = head
for (k = 0; k < i-1; k++)
    pre = pre.next
Vertex del = pre.next, aft = del.next
pre.next = aft // bypass del
delete del
```

1x

About Team Terms of use Privacy Policy



## Preguntas de Reflexion

¿Qué sucede con los punteros cuando se inserta o elimina un nodo?

Apuntan a un nuevo nodo ya sea que se elimine o inserte siguen manteniendo su puntero hacia algún nodo

¿Cómo afecta la posición de un nodo (inicio, medio, final) al tiempo de búsqueda?

En el tiempo en que la encuentre, ya que recorre la lista empezando por la cabeza y al estar mas cerca de la cabeza el nodo que queremos buscar lo encontrara más rápido, y al estar más lejos tardara más.

¿Qué ventajas tiene recorrer una lista enlazada frente a otras estructuras como arreglos?

Que sigue un orden, puedes eliminar un valor en especifico y es dinámica, ósea que puede crecer a como se vaya necesitando.

¿Cómo podrías implementar la comprobación de una lista vacía en un lenguaje de programación como Java?

Validando si mi lista es diferente o igual a nula y si usamos un isEmpty verificaríamos también si es empty es verdadero