

Universidad Tecnológica del Norte de Guanajuato

Tecnologías de la Información y Comunicación

Licenciatura en Ingeniería en Tecnologías de la Información e
Innovación Digital

Estructura de Datos

Alumno: Rojas Galindo Marlon

No. Control: 1224100711

Grupo: GTID-0141

Unidad II: Estructuras de Datos Básicas

Tema: Manipulación de Pila en VisuAlgo.net

Docente: Barrón Rodríguez Gabriel

Dolores Hidalgo, C.I.N., 20 de octubre de 2025

Visualgo.net interface showing a Singly Linked List (LL) with 5 nodes. The nodes contain values 65, -69, 81, -32, and -21. The head pointer is labeled 'head/0' and the tail pointer is labeled 'tail/4'. The interface includes a sidebar with options: Create(A), Peek, Push, Pop. The main area displays the list structure. The bottom status bar shows '1x' and navigation controls.

```
graph TD; head((head/0)) --> node0((65)); node0 --> node1((-69)); node1 --> node2((81)); node2 --> node3((-32)); node3 --> node4((-21)); node4 --> tail((tail/4));
```

Visualgo.net interface showing the same Singly Linked List (LL) with 5 nodes. The nodes contain values 65, -69, 81, -32, and -21. The head pointer is labeled 'head/0' and the tail pointer is labeled 'tail/4'. The interface includes a sidebar with options: Create(A), Peek, Push, Pop. The main area displays the list structure. The bottom status bar shows '1x' and navigation controls.

```
graph TD; head((head/0)) --> node0((65)); node0 --> node1((-69)); node1 --> node2((81)); node2 --> node3((-32)); node3 --> node4((-21)); node4 --> tail((tail/4));
```

Visualgo.net / en / list LL STACK QUEUE DLL DEQUE

Exploration Mode LOGIN

```
graph TD; head[head/0] --> 70((70)); 70 --> 65((65)); 65 --> -69((-69)); -69 --> 81((81)); 81 --> -32((-32)); -32 --> -21((-21)); -21 --> tail[tail/5];
```

Push 70 at top (head)

```
head points to vtx.  
Vertex vtx = new Vertex(v)  
vtx.next = head  
head = vtx
```

1x

About Team Terms of use Privacy Policy

Visualgo.net / en / list LL STACK QUEUE DLL DEQUE

Exploration Mode LOGIN

```
graph TD; head[head/0] --> 70((70)); 70 --> 65((65)); 65 --> -69((-69)); -69 --> 81((81)); 81 --> -32((-32)); -32 --> -21((-21)); -21 --> tail[tail/5];
```

Push 70 at top (head)

```
head points to vtx.  
Vertex vtx = new Vertex(v)  
vtx.next = head  
head = vtx
```

Create(A)
Peek
Push
Pop

v = 30 Go

1x

About Team Terms of use Privacy Policy

Visualgo.net interface showing a linked list with nodes: 30 (head), 70, 65, -69, 81, -32, -21 (tail). The list is titled "Push 30 at top (head)". The code block on the right shows:

```
head points to vtx.  
Vertex vtx = new Vertex(v)  
vtx.next = head  
head = vtx
```

Navigation buttons: Create(A), Peek, Push, Pop. The list is currently empty.

Visualgo.net interface showing a linked list with nodes: 30 (head), 70, 65, -69, 81, -32, -21 (tail). The list is titled "Push 30 at top (head)". The code block on the right shows:

```
head points to vtx.  
Vertex vtx = new Vertex(v)  
vtx.next = head  
head = vtx
```

Navigation buttons: Create(A), Peek, Push, Pop. The list is currently empty.

Visualgo.net interface showing a linked list with nodes: 70 (head/0), 65, -69, 81, -32, -21 (tail/5). The list is displayed vertically. The right sidebar shows the "Remove i = 0 (Head)" operation with the following code:

```
Delete tmp, which was the (previous) head

if empty, do nothing
tmp = head
head = head.next
delete tmp
```

The bottom of the interface includes a zoom slider (1x) and navigation controls.

Visualgo.net interface showing the same linked list. The left sidebar shows the "Create(A)" operation with the following code:

```
Create(A)
Peek
Push
Pop
```

The right sidebar shows the "Remove i = 0 (Head)" operation with the following code:

```
Delete tmp, which was the (previous) head

if empty, do nothing
tmp = head
head = head.next
delete tmp
```

The bottom of the interface includes a zoom slider (1x) and navigation controls.

Visualgo.net interface showing a linked list with nodes: 65 (head/0), -69, 81, -32, -21 (tail/4). The interface includes a sidebar with navigation options (STACK, QUEUE, DLL, DEQUE) and a bottom control bar with playback buttons and a 1x zoom level.

Remove i = 0 (Head)

```
Delete tmp, which was the (previous) head

if empty, do nothing
tmp = head
head = head.next
delete tmp
```

Visualgo.net interface showing the same linked list. The interface includes a sidebar with navigation options (STACK, QUEUE, DLL, DEQUE) and a bottom control bar with playback buttons and a 1x zoom level.

Peek top (head)

```
Return the value stored at the head: 65

if empty, return NOT_FOUND
return head.item
```

Left Sidebar:

- Create(A)
- Peek
- Push
- Pop

Visualgo

en /list LL STACK QUEUE DLL DEQUE

Exploration Mode LOGIN

65

head/0

-69

81

-32

-21

tail/4

Peek top (head)

Return the value stored at the head: 65.

if empty, return NOT_FOUND

return head.item

1x

⏮ ⏪ ⏸ ⏩ ⏭

About Team Terms of use Privacy Policy

Pilas con VisuAlgo

17-10-25

Marlon Rojas Galindo

¿Que estructura se forma visualmente con los nodos?

Se observan nodos ordenados verticalmente, cada nodo apunta al siguiente nodo, ~~que es~~ de arriba hacia abajo

¿Que sucede si haces pop en una lista vacia?

No realiza nada

23-01-21 10/10/21
¿Que diferencia hay entre
push y pop?

Que la operación push
agrega un elemento hasta
arriba de la pila y una
operación pop elimina el
ultimo elemento agregado.

Mencionar al menos 5
casos practicos en donde
se utilizen las pilas.

- Al momento de deshacer un
cambio en un software.

Al momento de mirar photos

- Recursividad. Aplicar la recursividad implica apilar procesos de una función para posteriormente realizarlos y descartarlos desde el último proceso hasta el primero.

- Al momento de llamar una función dentro de otra función.

- Historial. Guardar la última página visitada.