



## **TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN**

**Licenciatura en Ingeniería en Tecnologías de la Información e Innovación Digital (DSM)**

### **Desarrollo de software Multiplataforma**

#### **Estructura de datos**

Nearpod: Pilas

#### **UNIDAD II**

**Grupo: GTID141**

**Alumna:** Serna Rodríguez Sara Lizbeth

1224100715

## **Nearpod de Pilas**

A screenshot of a Nearpod presentation slide. The slide has a dark background with the word "Pilas" in large, light-colored serif font. On the left and right sides, there are white vertical panels with blue circular arrows pointing left and right respectively. At the bottom, a dark bar shows "2 of 15" and "Open navigator". A browser toolbar at the very bottom includes icons for Back, Forward, Stop, Refresh, and various extensions. A top navigation bar shows the URL "https://app.nearpod.com/presentation/pon+YD0XK" and a profile icon.

A screenshot of a Nearpod presentation slide. The slide has a dark background with the text "Recordar Tema de Listas" in large, light-colored serif font. On the left and right sides, there are white vertical panels with blue circular arrows pointing left and right respectively. At the bottom, a dark bar shows "3 of 15" and "Open navigator". A browser toolbar at the very bottom includes icons for Back, Forward, Stop, Refresh, and various extensions. A top navigation bar shows the URL "https://app.nearpod.com/presentation/pon+YD0XK" and a profile icon.

https://app.nearpod.com/presentation?pin=YCD0R

# Nodo Lista Simple

```
public class Nodo {  
    private int dato;  
    private Nodo siguiente;  
  
    public Nodo(int dato) {  
        this.dato = dato;  
        this.siguiente = null;  
    }  
  
    public int getDato() { return dato; }  
    public Nodo getSiguiente() { return siguiente; }  
    public void setSiguiente(Nodo siguiente) { this.siguiente = siguiente; }  
}
```

4 of 15 Open navigator ▾

Personal 1234560715.sor@gmail.com La sincronización está activada Configurar un nuevo perfil personal O Otros perfiles

https://app.nearpod.com/presentation?pin=YCD0R

# Lista Simple

```
public class ListSimple {  
    private Nodo cabeza;  
  
    public ListSimple() {  
        cabeza = null;  
    }  
  
    public void insertarInicio(int dato) {  
        Nodo nuevo = new Nodo(dato);  
        nuevo.setSiguiente(cabeza);  
        cabeza = nuevo;  
    }  
  
    public void mostrar() {  
        Nodo actual = cabeza;  
        while (actual != null) {  
            System.out.print(actual.getDato());  
            actual = actual.getSiguiente();  
        }  
    }  
}
```

5 of 15 Open navigator ▾

Personal 1234560715.sor@gmail.com La sincronización está activada Configurar un nuevo perfil personal O Otros perfiles

https://app.nearpod.com/presentations/pin/10008

Question 1 / 10

¿Cuál es la principal función de la clase Nodo en una lista simple en Java?

A. Almacenar un elemento y enlazarlo con el siguiente.

B. Eliminar elementos de la lista.

C. Ordenar los elementos de la lista.

D. Almacenar múltiples elementos en un solo Nodo.

Personal  
1234567890abc@gmail.com  
● La incorporación está activada

Configurar un nuevo perfil personal.  
Otros perfiles

Next

8 of 15 Open navigator 12:00 a.m. 07/11/2025

https://app.nearpod.com/presentations/pin/10008

Question 2 / 10

¿Qué atributo debe tener la clase Nodo para referenciar al siguiente Nodo en la lista?

A. Un atributo de tipo String llamado 'siguiente'.

B. Un atributo de tipo entero llamado 'siguiente'.

C. Un atributo de tipo Nodo llamado 'siguiente'.

D. Un atributo de tipo booleano llamado 'siguiente'.

Personal  
1234567890abc@gmail.com  
● La incorporación está activada

Configurar un nuevo perfil personal.  
Otros perfiles

Back Next

1 answer(s) selected

8 of 15 Open navigator 12:00 a.m. 07/11/2025

https://app.nearpod.com/presentations/pro/YQ8R8

Question 3 / 10

¿Qué tipo de acceso se recomienda para los atributos de la clase Nodo?

A. Sin especificador de acceso.

B. Protegido.

C. Privado.

D. Público.

Personal  
1234567890abc@gmail.com  
La imprecisión está activada

Configurar un nuevo perfil personal  
Otro perfil

Back Next

6 of 15 Open navigator

https://app.nearpod.com/presentations/pro/YQ8R8

Question 4 / 10

¿Cómo se puede crear un nuevo Nodo en Java?

A. Declarando un Nodo sin inicializarlo.

B. No se puede crear un Nodo en Java.

C. Usando el operador 'new' para crear una instancia de Nodo.

D. Usando un método estático para crear un Nodo.

Personal  
1234567890abc@gmail.com  
La imprecisión está activada

Configurar un nuevo perfil personal  
Otro perfil

Back Next

6 of 15 Open navigator

Question 5 / 10

¿Qué método se podría implementar en la clase Nodo para obtener el valor almacenado?

A. Un método llamado 'eliminarValor'.

B. Un método llamado 'setValor'.

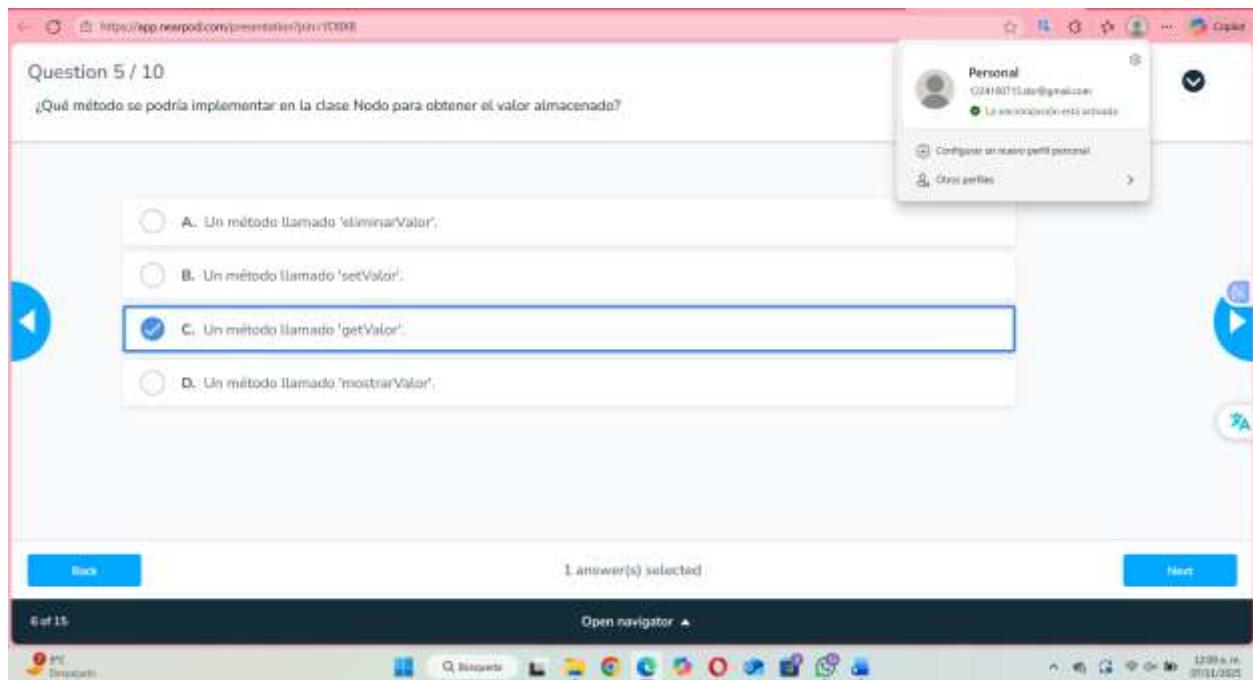
C. Un método llamado 'getValor'.

D. Un método llamado 'mostrarValor'.

Back Next

1 answer(s) selected

6 of 15 Open navigator ▾



Question 6 / 10

¿Qué se necesita para enlazar un nuevo Nodo al final de una lista simple?

A. Agregar el nuevo Nodo directamente al inicio.

B. No se necesita hacer nada, el nuevo Nodo se agrega automáticamente.

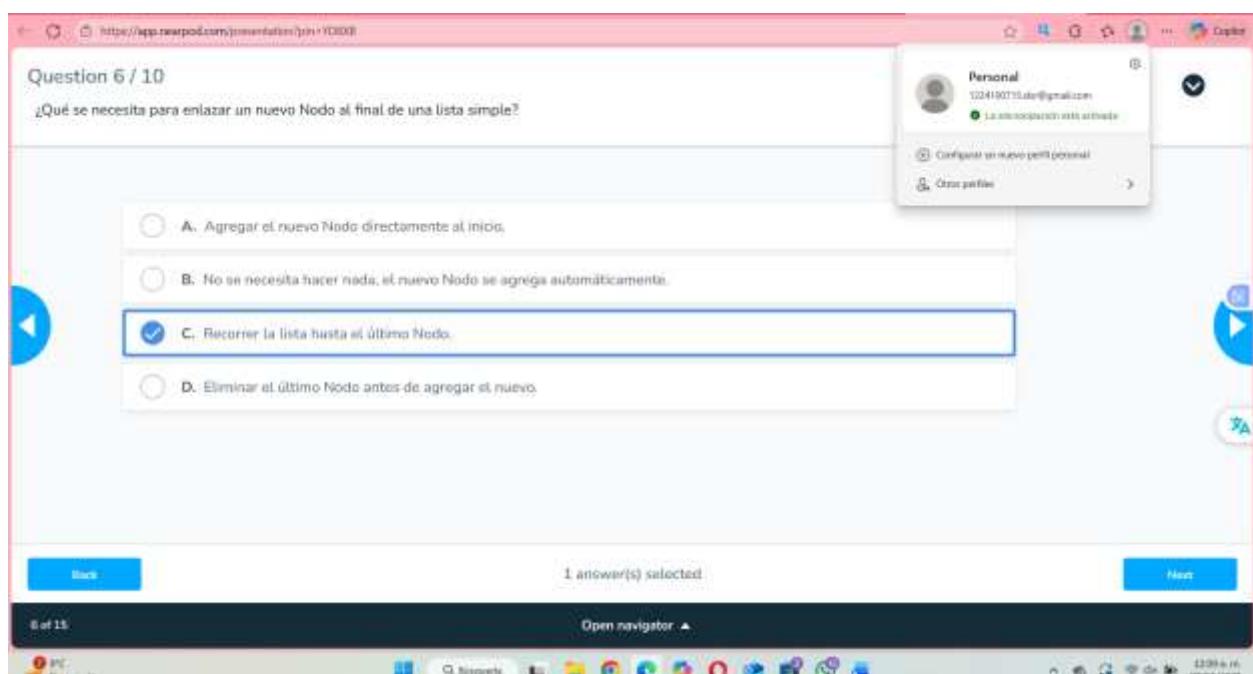
C. Recorrer la lista hasta el último Nodo.

D. Eliminar el último Nodo antes de agregar el nuevo.

Back Next

1 answer(s) selected

6 of 15 Open navigator ▾



https://app.nearpod.com/presentations/jm+YD00B

Question 7 / 10

¿Cuál es una desventaja de usar listas simples en comparación con listas dobles?

A. No permite almacenar elementos duplicados.

B. Es más difícil de implementar.

C. Ocupa más memoria que una lista doble.

D. No se puede acceder a los elementos en ambas direcciones.

Back Next

1 answer(s) selected

6 of 15 Open navigator

12:09 a.m. 30/11/2023

https://app.nearpod.com/presentations/jm+YD00B

Question 8 / 10

¿Qué se debe hacer al eliminar un Nodo de una lista simple?

A. Eliminar todos los Nodos de la lista.

B. Eliminar el Nodo sin ajustar las referencias.

C. Ajustar las referencias de los Nodos adyacentes.

D. No se puede eliminar un Nodo de una lista simple.

Back Next

1 answer(s) selected

6 of 15 Open navigator

12:10 a.m. 30/11/2023

Question 9 / 10

¿Qué constructor se recomienda para la clase Nodo?

A. Un constructor que acepte solo un valor entero.

B. Un constructor que no acepte parámetros.

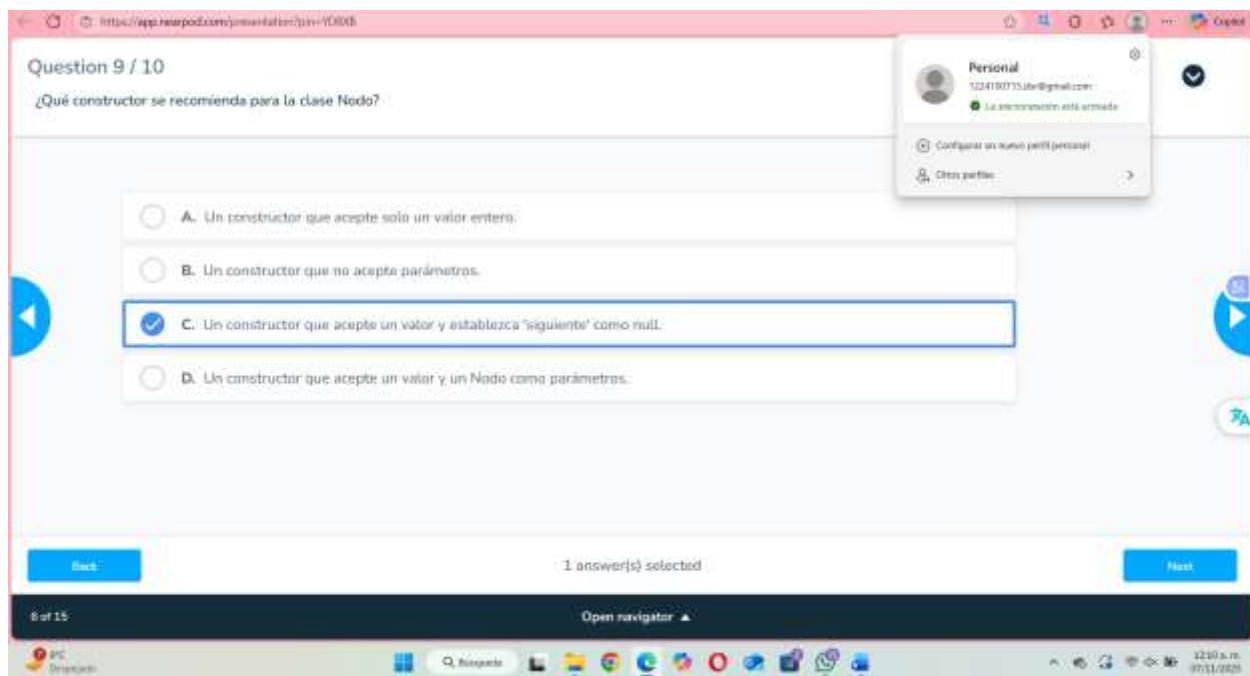
C. Un constructor que acepte un valor y establezca "siguiente" como null.

D. Un constructor que acepte un valor y un Nodo como parámetros.

Back Next

1 answer(s) selected

6 of 15 Open navigator ▾



Question 10 / 10

¿Qué tipo de estructura de datos es una lista simple?

A. Estructura de datos jerárquica.

B. Estructura de datos estática.

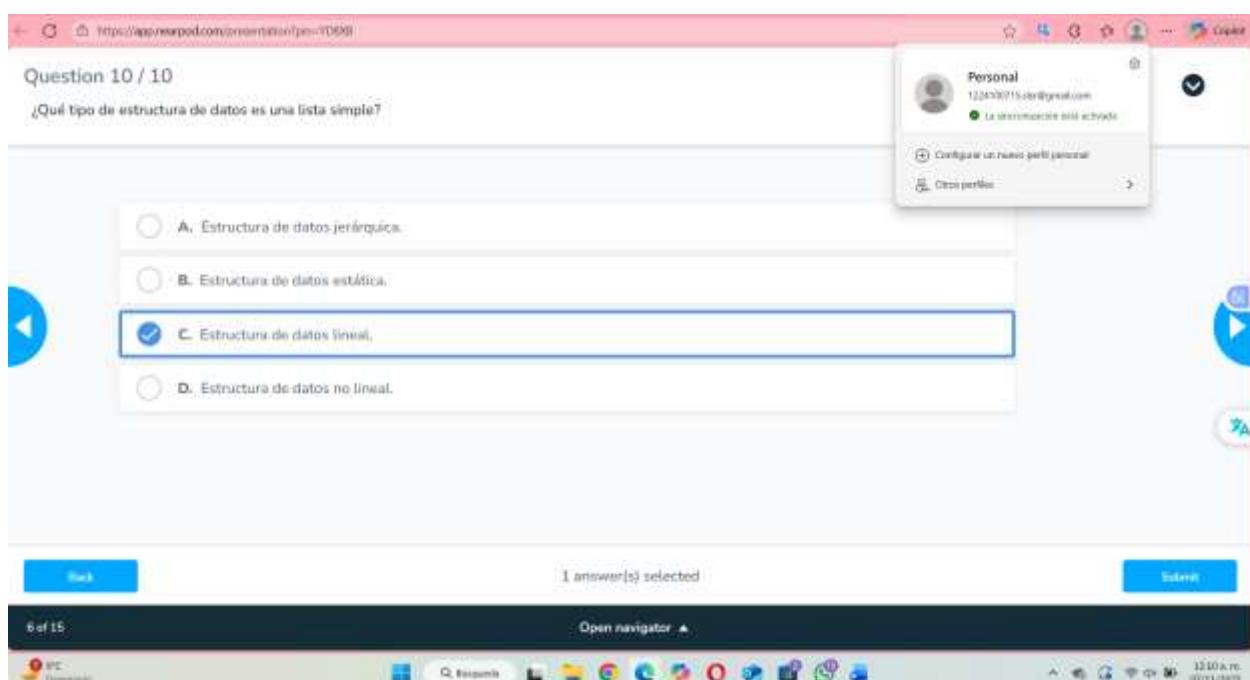
C. Estructura de datos lineal.

D. Estructura de datos no lineal.

Back Submit

1 answer(s) selected

6 of 15 Open navigator ▾



https://app.nearpod.com/presentation?pn=10300

## Listas y Pilas

YOU SCORED 10 OUT OF 10

100% CORRECT  
0% INCORRECT  
0% NO ANSWER

Personal  
G20180715.usy@gmail.com  
La autenticación está activada

Configurar un nuevo perfil personal  
otras pestañas

My Answers

¿Cuál es la principal función de la clase Nodo en una lista simple en Java?

Almacenar un elemento y enlazarlo con el siguiente. ✓

6 of 15 Open navigator ▾

PC Unreadable

12:11 a.m. 07/11/2023

https://app.nearpod.com/presentation?pn=10300

## PILA

Una pila es una estructura de datos tipo LIFO (last in first out, último en entrar primero en salir) en la que los datos (todos del mismo tipo) se añaden y se eliminan por el mismo extremo, denominado cima de la pila.

The diagram illustrates a stack (PILA) with five green rectangular boxes stacked vertically. The top box contains the number '50' and is labeled 'CIMA' (Top) with an upward arrow to its right. The bottom box contains the number '10' and is labeled 'ELIMINAR' (Delete) with a downward arrow to its left. The other three boxes contain the numbers '40', '30', and '20' from top to bottom. To the left of the stack is a downward-pointing yellow arrow labeled 'AÑADIR' (Add). The entire slide has a black border.

7 of 15 Open navigator ▾

PC Unreadable

12:11 a.m. 07/11/2023

**2. Operaciones Básicas**

Operación	Descripción	Método
push()	Inserta un elemento en la cima.	pila.push("x")
pop()	Elimina el elemento superior.	pila.pop();
peek()	Devuelve el elemento superior sin eliminarlo.	pila.peek();
isEmpty()	Verifica si está vacía.	pila.isEmpty();

8 of 15 Open navigator ▾

Personal 1234567890123456@gmail.com La conexión está activada Configurar un nuevo perfil personal Otros perfiles

# Implementación con Arreglos

```
public class PilaArreglo {  
    private int[] pila;  
    private int topo;  
    private int capacidad;  
  
    // Constructor vacío  
    public PilaArreglo() {  
        this.pila = new int[10];  
        this.topo = -1;  
        this.capacidad = 10;  
    }  
  
    // Insertar elemento  
    public void push(int dato) {  
        if (topo <= capacidad - 1) {  
            pila[topo + 1] = dato;  
            topo++;  
        } else {  
            System.out.println("Pila llena");  
        }  
    }  
  
    // Eliminar elemento  
    public void pop() {  
        if (topo == -1) {  
            System.out.println("Pila vacía");  
        } else {  
            topo--;  
            System.out.println("Elemento eliminado: " + pila[topo]);  
        }  
    }  
}
```

9 of 15 Open navigator ▾

Personal 1234567890123456@gmail.com La conexión está activada Configurar un nuevo perfil personal Otros perfiles

The screenshot shows a presentation slide with a dark background. At the top, the title 'Implementación con ArrayList' is displayed in large, bold, white font. Below the title is a code block in Java:

```
import java.util.ArrayList;

class ArrayList {
    private ArrayList<T> elementos = new ArrayList<T>();

    public void add(T valor) {
        elementos.add(valor);
    }

    public T pop() {
        if (isEmpty()) return null;
        return elementos.remove(elementos.size() - 1);
    }

    public T peek() {
        if (isEmpty()) return null;
        return elementos.get(elementos.size() - 1);
    }

    public boolean isEmpty() {
        return elementos.isEmpty();
    }
}
```

At the bottom of the slide, there is a navigation bar with '10 of 15' on the left and 'Open navigator ▾' on the right. The browser's address bar shows the URL <https://app.rewpod.com/presentations/pin/1C00>. The browser interface includes various icons and a status bar at the bottom.

The screenshot shows a presentation slide with a dark background. The title 'Aplicaciones En Programación' is centered at the top in large, bold, white font. Below the title, there are five numbered sections with corresponding icons and bullet points:

- 1. Evaluación de expresiones matemáticas**
  - Se usan pilas para evaluar expresiones en notación posfixa (RPN) o infix.
  - Ejemplo: convertir  $2 + 4 * 2$  en posfixa y evaluarla usando una pila.
- 2. Conversión entre notaciones**
  - De infix a posfix o prefix, utilizando pilas para operadores y operandos.
- 3. Verificación de paréntesis balanceados**
  - Se aplican los símbolos de apertura y se desapilas al encontrar cierres.
  - Util en compiladores y editores de código.
- 4. Recursividad**
  - Cada llamada recursiva se apila en la pila de ejecución.
  - Ejemplos: funciones como factorial, Fibonacci, recorrido DFL.
- 5. Algoritmos de búsqueda en grafos (DFS)**
  - El recorrido en profundidad (Depth First Search) usa una pila para explorar nodos.

At the bottom of the slide, there is a navigation bar with '11 of 15' on the left and 'Open navigator ▾' on the right. The browser's address bar shows the URL <https://app.rewpod.com/presentations/pin/1D6E>. The browser interface includes various icons and a status bar at the bottom.

Aplicaciones en Programación

**6. Deshacer/rehacer en editores**

- Cada acción se guarda en una pila para permitir undo/redo.

**7. Manejo de llamadas a funciones**

- El sistema usa una pila para almacenar el contexto de cada función activa.

**8. Evaluación de expresiones booleanas o lógicas**

- Similar a las matemáticas, pero con operadores lógicos (AND, OR, NOT).

**9. Compiladores e intérpretes**

- Para análisis sintáctico, control de bloques, y ejecución de instrucciones.

12 of 15 Open navigator ▾

Question 1 / 13

¿Cuál es la principal desventaja de usar un arreglo para implementar una pila?

A. Su complejidad.

B. Que no permite duplicados.

C. Que no permite recorrer sus elementos.

D. Que tiene tamaño fijo y puede distorsionarse.

1 answer(s) selected

Next

13 of 15 Open navigator ▾

Question 2 / 13

Completa la condición para evitar el desbordamiento de la pila:

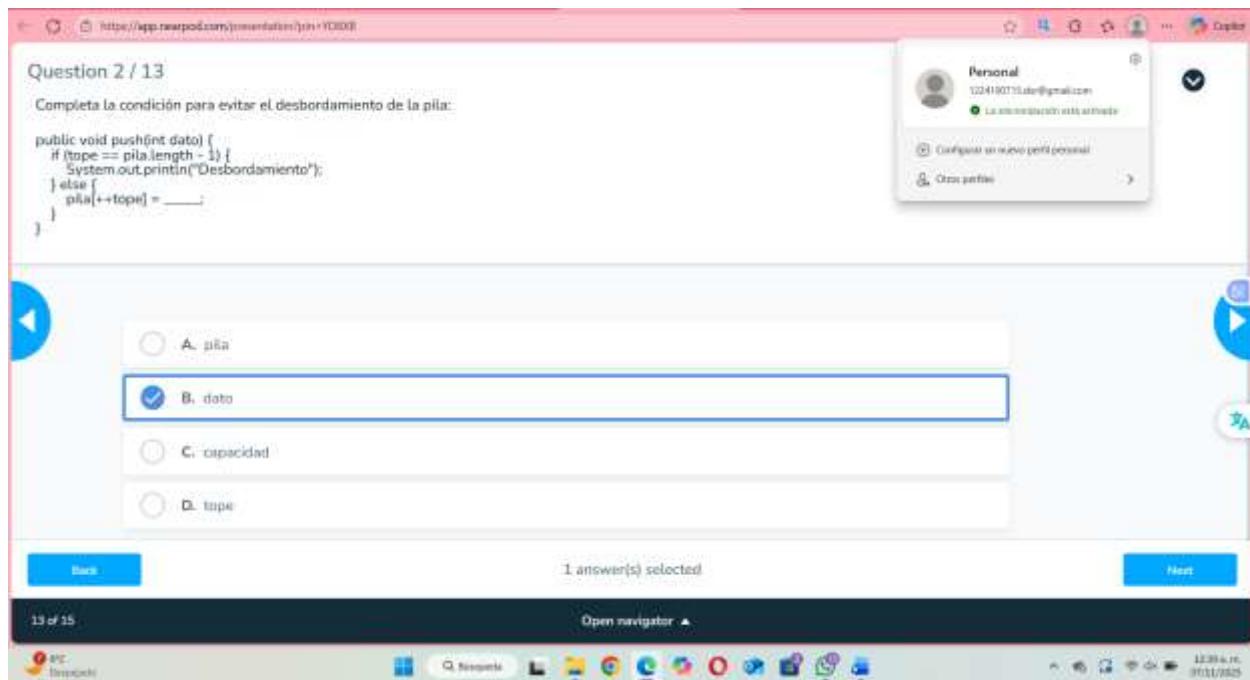
```
public void push(int dato) {  
    if (topo == pila.length - 1) {  
        System.out.println("Desbordamiento");  
    } else {  
        pila[++topo] = ____;  
    }  
}
```

A. pila  
 B. dato  
 C. capacidad  
 D. topo

1 answer(s) selected

Back Next

13 of 15 Open navigator



Question 3 / 13

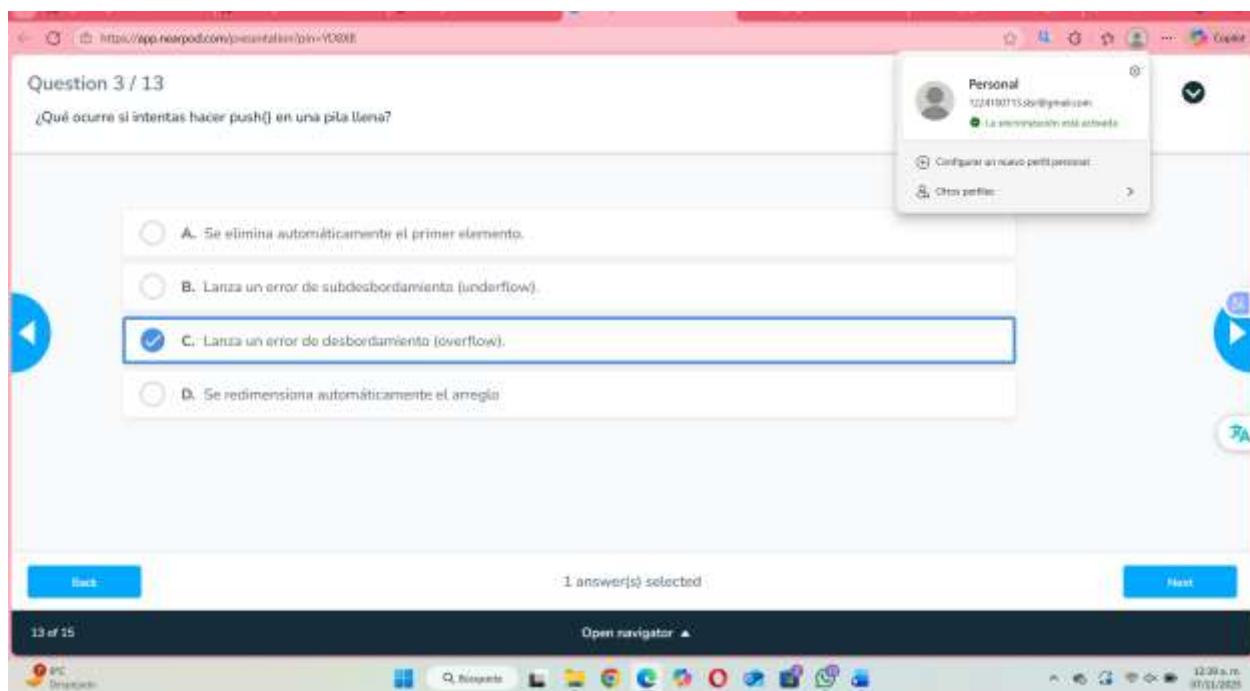
¿Qué ocurre si intentas hacer push() en una pila llena?

A. Se elimina automáticamente el primer elemento.  
 B. Lanza un error de subdesbordamiento (underflow).  
 C. Lanza un error de desbordamiento (overflow).  
 D. Se redimensiona automáticamente el arreglo.

1 answer(s) selected

Back Next

13 of 15 Open navigator



Question 4 / 13

Completa el método main para insertar y mostrar elementos:

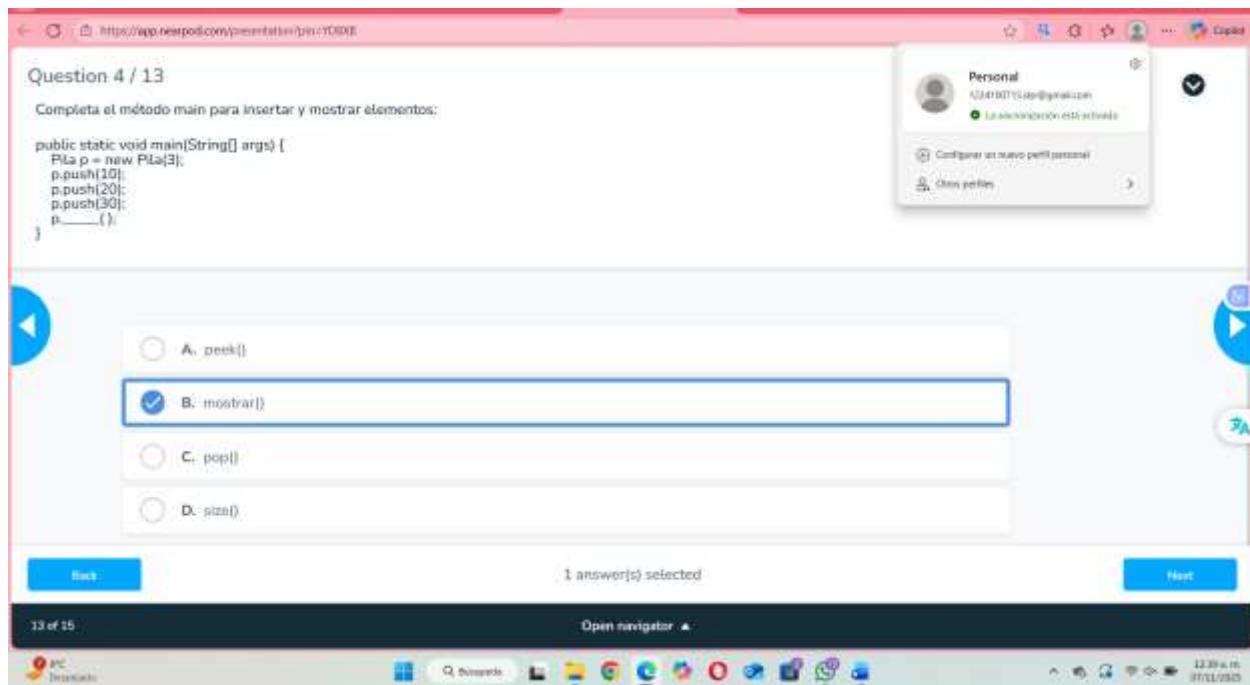
```
public static void main(String[] args) {  
    Pila p = new Pila[3];  
    p.push(10);  
    p.push(20);  
    p.push(30);  
    p_____();  
}
```

A. peek()  
 B. mostrar()  
 C. pop()  
 D. size()

1 answer(s) selected

Back Next

13 of 15 Open navigator



Question 5 / 13

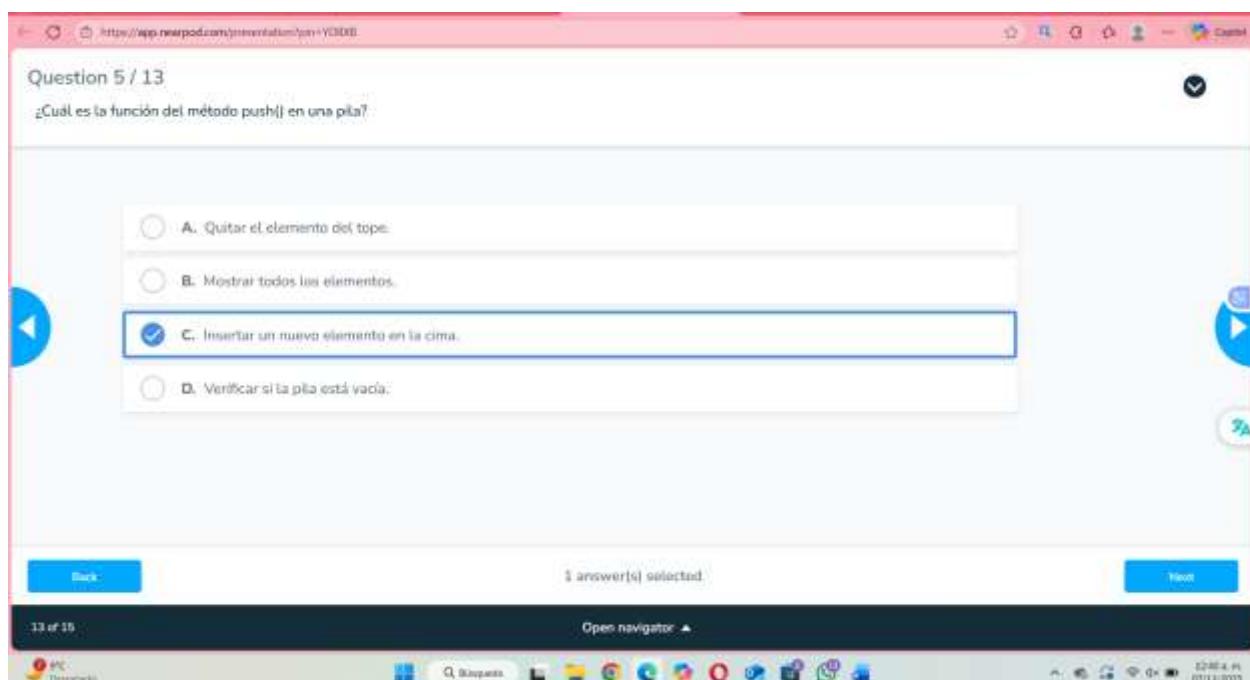
¿Cuál es la función del método push() en una pila?

A. Quitar el elemento del topo.  
 B. Mostrar todos los elementos.  
 C. Insertar un nuevo elemento en la cima.  
 D. Verificarse si la pila está vacía.

1 answer(s) selected

Back Next

13 of 15 Open navigator



Question 6 / 13

Completa el código para eliminar el último elemento insertado:

```
public int pop() {
    if (isEmpty()) {
        System.out.println("Pila vacía");
        return -1;
    }
    return pila[_____];
```

A. 0  
 B. pila.length  
 C. tope  
 D. dato

1 answer(s) selected

13 of 15

Open navigator ▾

Question 7 / 13

¿Qué devuelve el método peek()?

A. El primer elemento de la pila.  
 B. El tamaño total de la pila.  
 C. El elemento superior sin eliminarlo.  
 D. Todos los elementos en orden inverso.

1 answer(s) selected

13 of 15

Open navigator ▾

Question 8 / 13

Completa la instrucción para obtener el elemento superior sin eliminarlo:

```
public int peek() {  
    if (isEmpty()) return -1;  
    return pila[_____];  
}
```

A. pila.length - 1  
 B. topo  
 C. 0  
 D. capacidad

1 answer(s) selected

Back Next

13 of 15 Open navigator ▾

PC Unacademy 12:40 a.m. 07/11/2025

Question 9 / 13

¿Qué método se utiliza para eliminar el elemento superior de la pila?

A. pop()

B. peek()

C. push()

D. remove()

Back

1 answer(s) selected

Next

13 of 15

Open navigator ▾

Question 10 / 13

Completa la condición que verifica si la pila está vacía:

```
public boolean isEmpty() {  
    return _____ == -1;  
}
```

A. pilalength

B. capacidad

C. tope

D. pila[0]

Back

1 answer(s) selected

Next

13 of 15

Open navigator ▾

Question 11 / 13

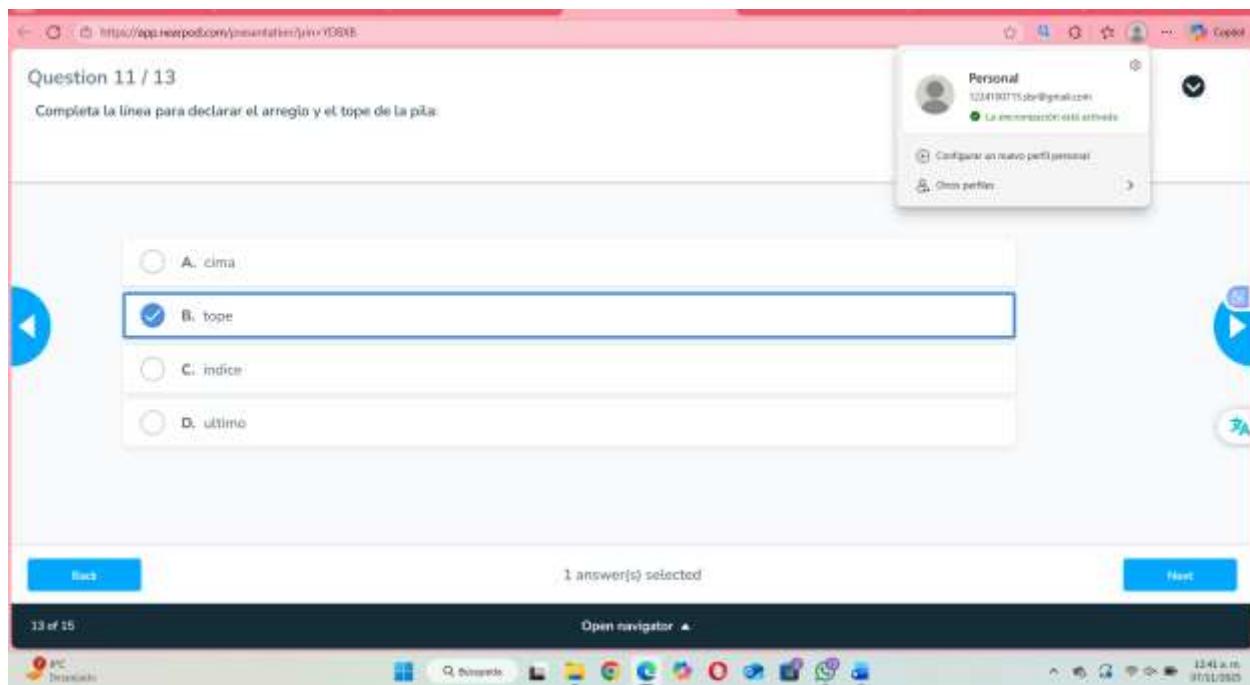
Completa la línea para declarar el arreglo y el tope de la pila:

A. cima  
 B. tope  
 C. indice  
 D. ultimo

Back Next

1 answer(s) selected

13 of 15 Open navigator ▾



Question 12 / 13

Completa el constructor de la clase para inicializar la pila y el tope:

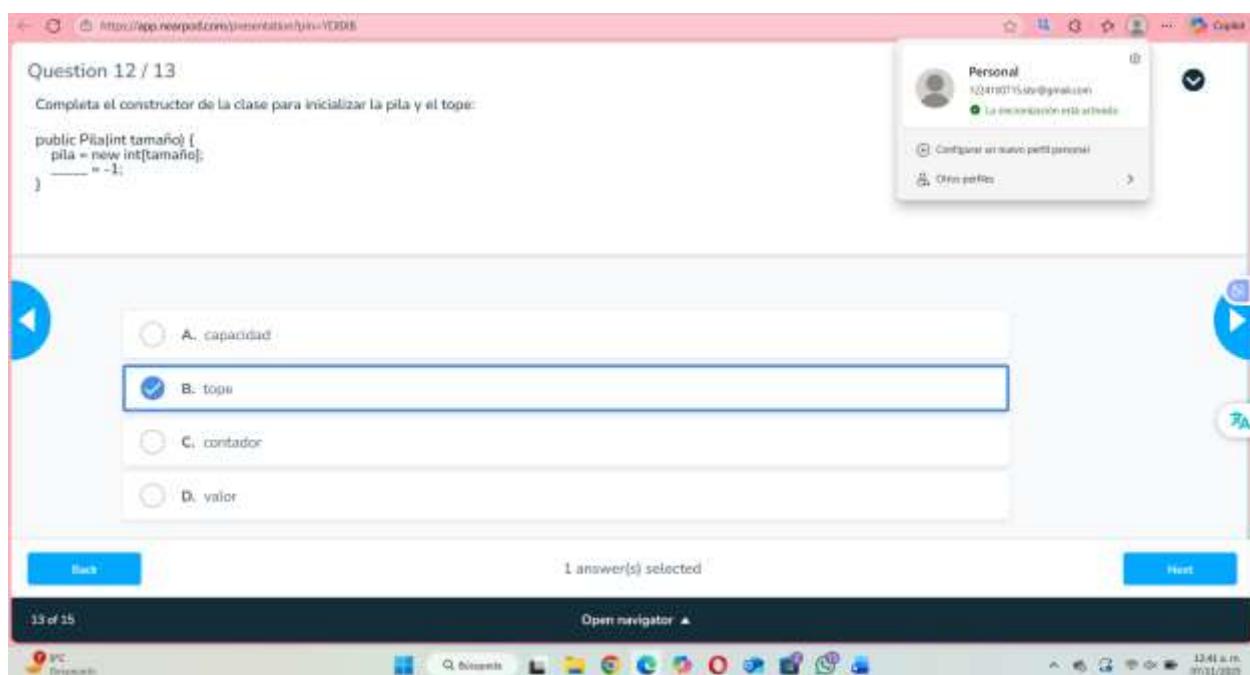
```
public Pilas(int tamaño) {  
    pila = new int[tamaño];  
    _____ = -1;  
}
```

A. capacidad  
 B. tope  
 C. contador  
 D. valor

Back Next

1 answer(s) selected

13 of 15 Open navigator ▾



Question 13 / 13

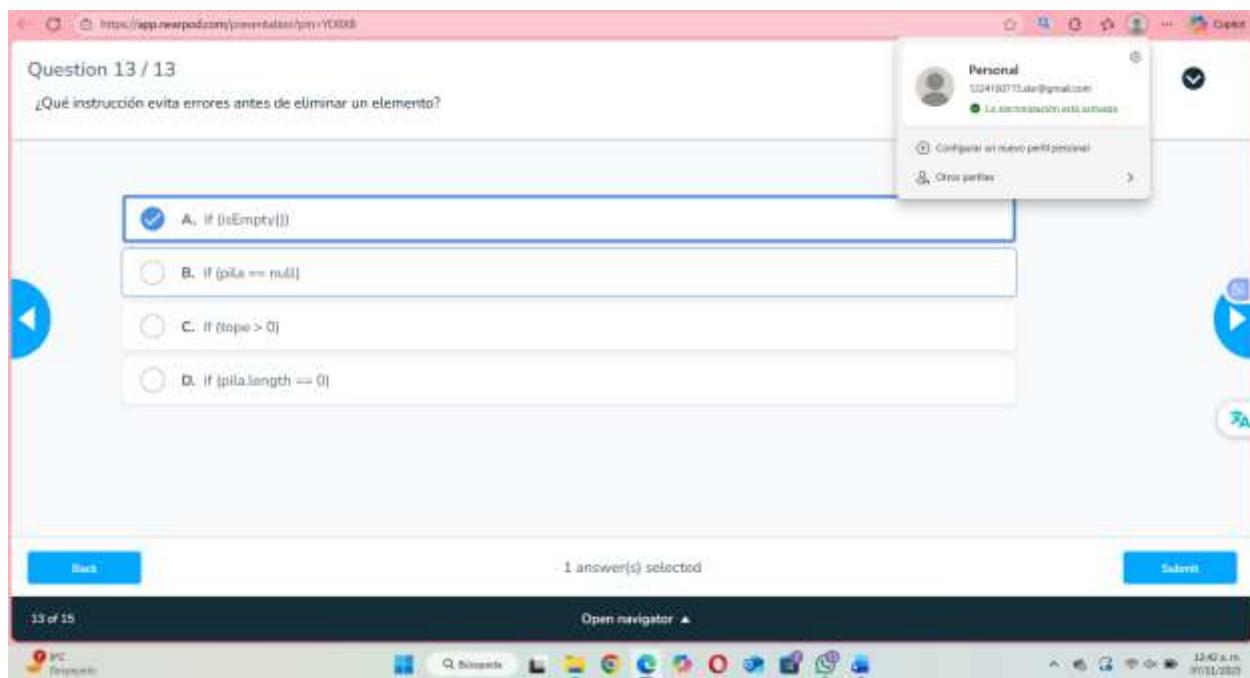
¿Qué instrucción evita errores antes de eliminar un elemento?

A. if (isEmpty())  
B. if (pila == null)  
C. if (topo > 0)  
D. if (pila.length == 0)

1 answer(s) selected

13 of 15

Open navigator ▾



Quiz de Pilas

YOU SCORED 13 OUT OF 13

100% CORRECT  
0% INCORRECT  
0% NO ANSWER

My Answers

¿Cuál es la principal desventaja de usar un arreglo para implementar una pila?

Su complejidad

13 of 15

Open navigator ▾

