

Técnicas de programación



Ph.D. Johnny Alexander Salazar Cardona





Introducción a las estructuras de datos

Estructuras de datos

¿Qué son?

Una estructura de datos es una **forma de organizar y almacenar información en la memoria de un computador para que pueda ser utilizada de manera eficiente.**

¿Por qué son importantes las estructuras de datos?

1. Permiten almacenar y gestionar información de manera eficiente.
2. Facilitan la búsqueda, inserción y eliminación de datos.
3. Son la base de la optimización en algoritmos y software.
4. Se usan en bases de datos, inteligencia artificial, videojuegos, redes, etc.

Ejemplo práctico:

Si se tiene una agenda de contactos en un teléfono, se necesita una estructura de datos que:

- Permita guardar los contactos.
- Permita buscar un contacto rápidamente.
- Permita eliminar o modificar un contacto.

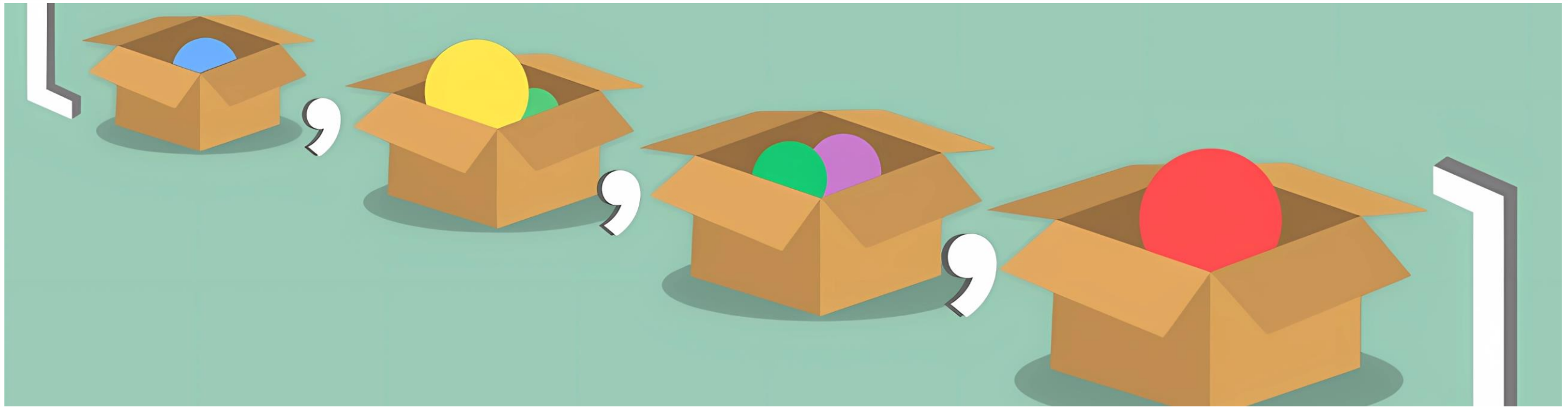
Estructuras de datos

¿Todos son iguales?

La estructura en la cual se almacenan los datos **depende en gran medida de lo que se desea realizar con los datos**.
Por ejemplo: Vamos a generar un algoritmo para gestionar tareas **¿Qué estructura debo utilizar?**

- Si se registran las tareas en cualquier orden → es como una lista.
- Si se apilan las tareas y se saca la ultima primero → es una pila.
- Si se atienden las tareas en orden de llegada → es una cola.

Tipo	Descripción	Ejemplo en la vida real
Listas	Colección ordenada de datos. Se puede acceder a cualquier elemento por su posición.	Lista de compras en papel.
Pilas (Stack)	Último en entrar, primero en salir (LIFO) (Last In, First Out).	Una pila de platos en un restaurante.
Colas (Queue)	Primero en entrar, primero en salir (FIFO) (First In, First Out).	Fila para pagar en un supermercado.
Arboles	Estructura jerárquica con nodos.	Un organigrama de empresa.
Grafos	Conjunto de nodos conectados por aristas.	Un mapa de rutas de autobús.



Introducción a listas

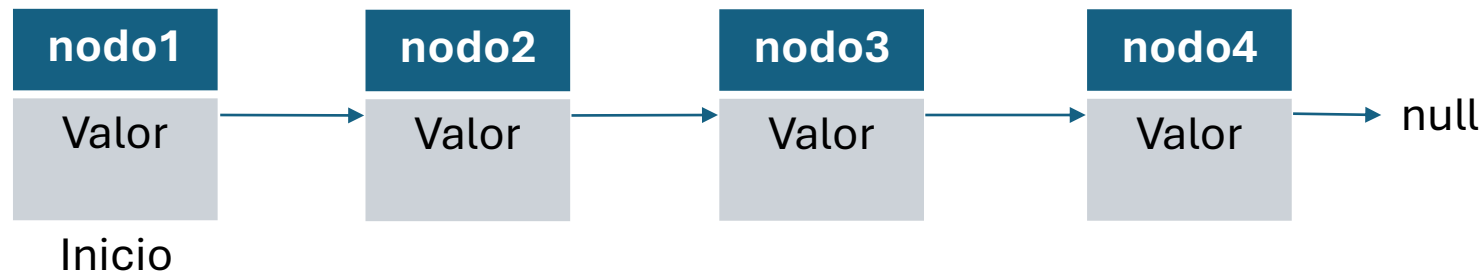
Listas

¿Qué son?

1. Una lista es una colección ordenada de elementos donde se pueden insertar, eliminar y recorrer datos
2. A diferencia de un arreglo, el tamaño de esta es dinámica y puede crecer o disminuir según la cantidad de elementos que almacene.
3. Puede ser simplemente enlazada o doblemente enlazada.

¿Qué pasa si elimino el elemento del nodo3?

¿Qué pasa si agrego un elemento entre el nodo 3 y 4?



Funciones típicas en una listas

¿Cómo se manipula?

Función	Descripción
Insertar (Agregar/Append)	Añade un elemento al final o en una posición específica.
Eliminar (Remove/Delete)	Borra un elemento por valor o por índice.
Buscar (Search/Find)	Encuentra un elemento en la lista y devuelve su posición.
Actualizar (Update/Modify)	Modifica un elemento en una posición específica.
Obtener (Get/Access)	Devuelve un elemento en una posición específica.
Recorrer (Iterate)	Recorre la lista y procesa cada elemento.
Ordenar (Sort)	Organiza los elementos en orden ascendente o descendente.
Reversar (Reverse)	Invierte el orden de los elementos de la lista.
Longitud (Size/Length)	Devuelve el número de elementos en la lista.
Vaciar (Clear)	Elimina todos los elementos de la lista.

Listas

Algunos ejemplos en pseudocodigo

EJEMPLO BASICO:

INICIO

DEFINIR lista COMO ENTERO[]

AGREGAR(lista, 10) // Agrega el número 10 a la lista

AGREGAR(lista, 20) // Agrega el número 20 a la lista

PARA i DESDE 0 HASTA LONGITUD(lista)-1 HACER

 ESCRIBIR lista[i]

FIN PARA

ELIMINAR(lista, 10) // Elimina el número 10 de la lista

FIN

OTRAS FUNCIONALIDADES:

Insertar un elemento en una posición específica:

- INSERTAR(lista, posicion, elemento)

Buscar un elemento y retornar su posición

FUNCION Buscar(lista, elemento)

 PARA i DESDE 0 HASTA LONGITUD(lista)-1 HACER

 SI lista[i] == elemento ENTONCES

 RETORNAR i

 FIN SI

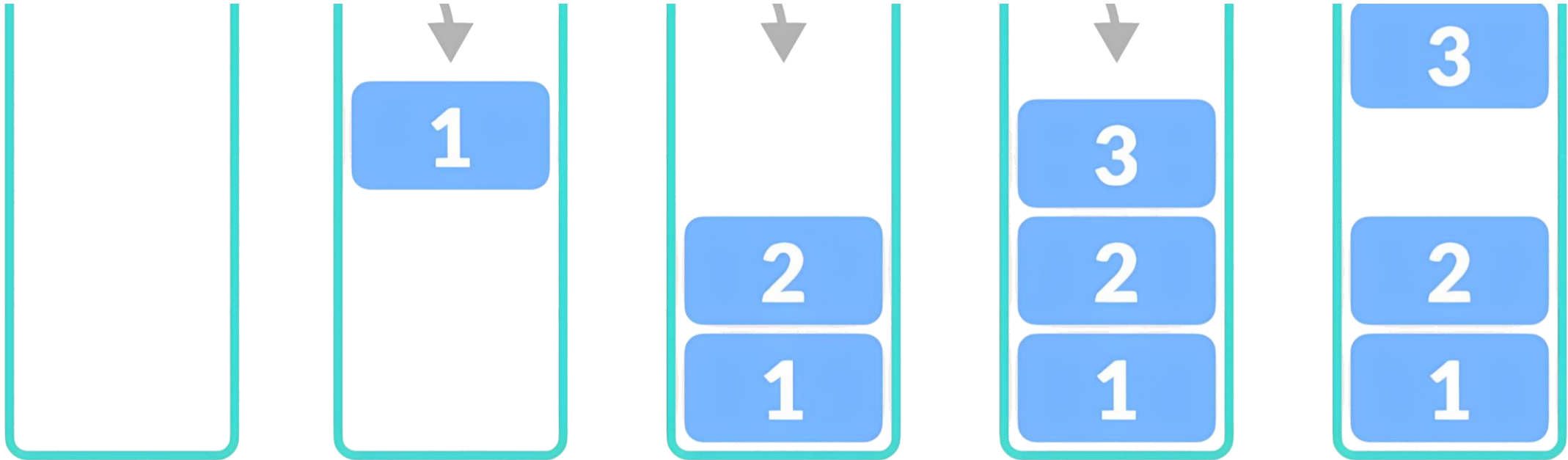
 FIN PARA

 RETORNAR -1

FIN FUNCION

Retornar elemento de una posición:

- RETORNAR lista[indice]

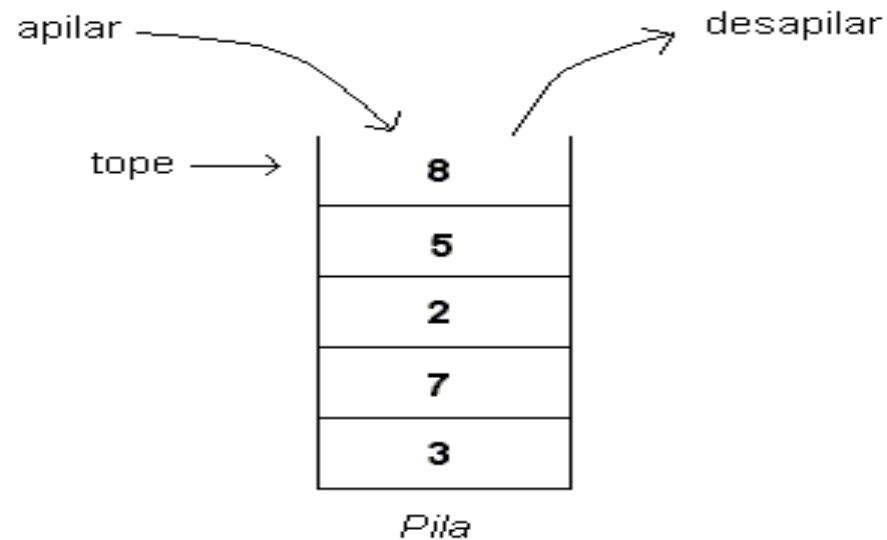


Introducción a pilas

Pilas

¿Qué son?

1. Una Pila (stack) es una lista de elementos de la cual **sólo se puede extraer el último elemento insertado**.
2. La posición en donde se encuentra dicho elemento se denomina **tope de la pila**.
3. También se conoce a las pilas como **listas LIFO (LAST IN - FIRST OUT: el último que entra es el primero que sale)**.



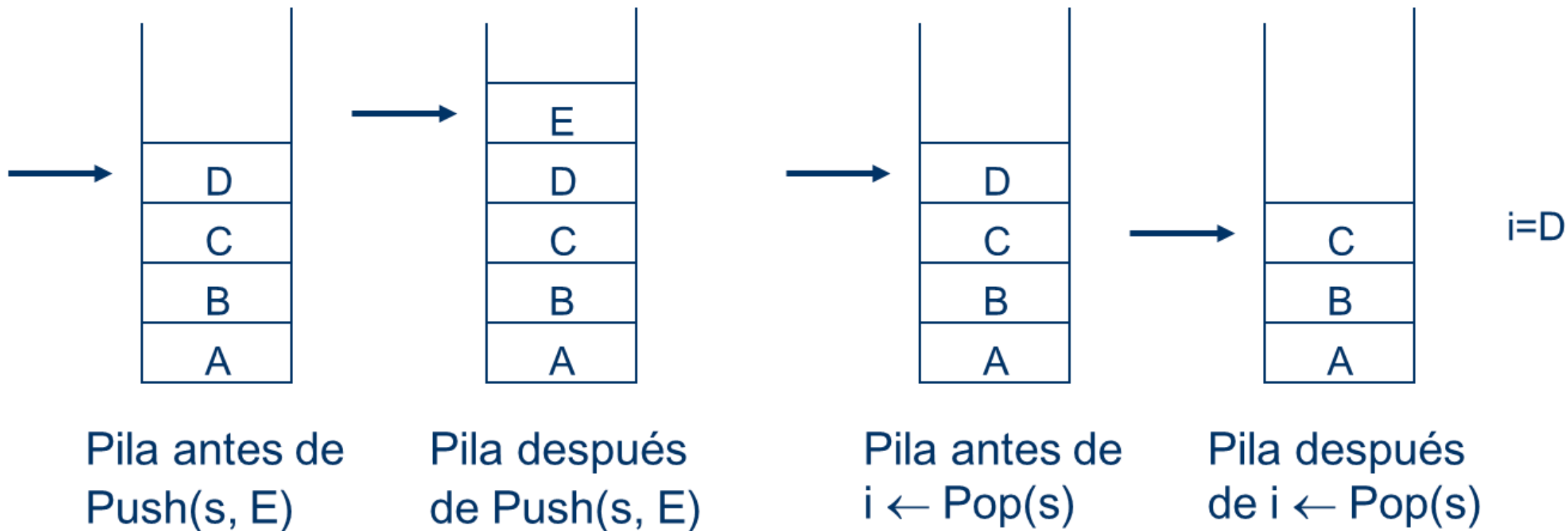
Pilas: Operaciones básicas

¿Cuáles son?

- **APILAR (push):** Inserta el elemento x en el tope de la pila (push en inglés).
- **DESAPILAR (pop):** retorna el elemento que se encuentre en el tope de la pila y lo elimina de ésta.
- **TOPE:** retorna el elemento que se encuentre en el tope de la pila, pero sin eliminarlo de ésta.
- **ESTA VACIA:** retorna verdadero si la pila no contiene elementos, falso en caso contrario.
- **BUSCAR:** Retorna la true si el elemento a buscar existe en la pila.
- **ELIMINAR:** Elimina la pila.
- **LISTAR:** Imprime los elementos de la pila.

Pilas: Ejemplo

Ejemplo



Pilas

Algunos ejemplos en pseudocodigo

EJEMPLO BASICO:

INICIO

DEFINIR pila COMO PILA ENTERO

APILAR(pila, 5)

APILAR(pila, 8)

APILAR(pila, 12)

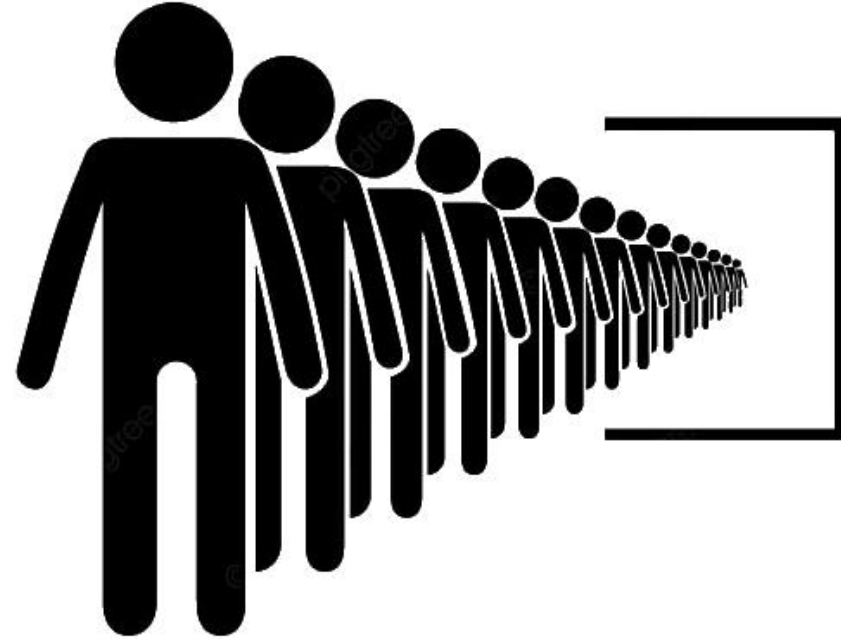
ELEMENTO \leftarrow DESAPILAR(pila)

ESCRIBIR "Elemento eliminado: ", ELEMENTO

ESCRIBIR "Elemento en la cima: ", CIMA(pila)

FIN



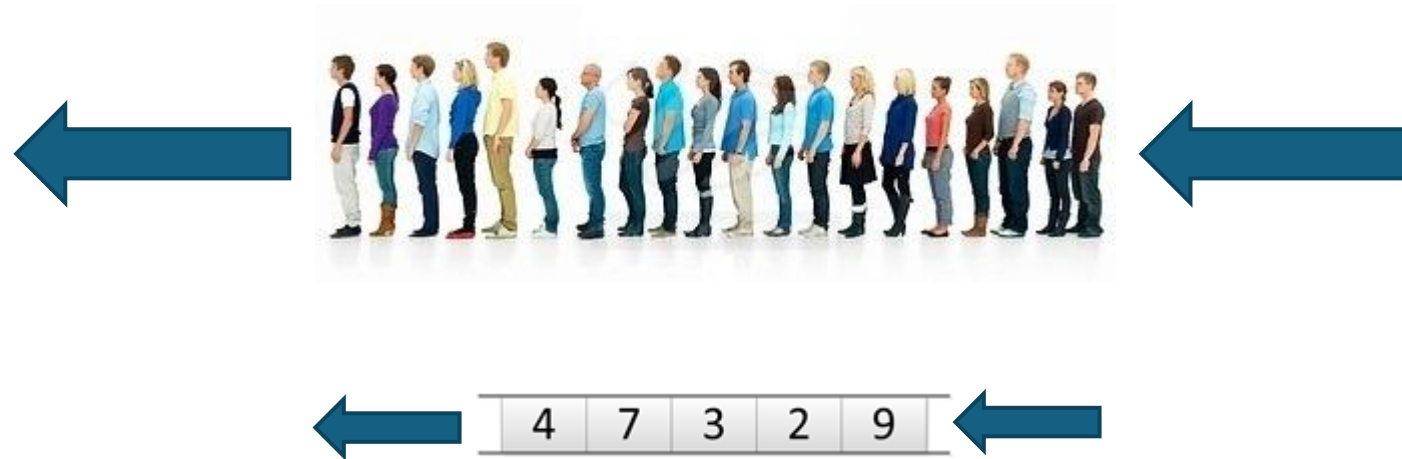


Introducción a colas

Colas

¿Qué son?

- Una cola es simplemente un lugar para almacenar datos, esos **datos se insertan unos detrás de otros y para extraer siempre se lo hace por adelante de la cola donde se encuentra el primer elemento. FIFO (First In First Out)**



Colas: Operaciones básicas

¿Cuáles son?

- **Encolar:** Añadir un elemento a la cola
- **Desencolar:** Eliminar un elemento de la cola
- **Esta vacía:** retorna verdadero si la pila no contiene elementos, falso en caso contrario.
- **Buscar:** Retorna la true si el elemento a buscar existe en la cola.
- **Eliminar:** Elimina la cola.
- **Listar:** Imprime los elementos de la cola.

Colas: Ejemplo

Ejemplo

Inicio

15	20	9	18	19
----	----	---	-------	----	----

Fin

1.Ejemplo de Cola

15	20	9	18	19
----	----	---	-------	----	----

2.Vamos a Insertar el 13 en la Cola.

15	20	9	18	19	13
----	----	---	-------	----	----	----

3.Sacamos el frente de la Cola (15)

20	9	18	19	13
----	---	-------	----	----	----

Colas

Algunos ejemplos en pseudocodigo

EJEMPLO BASICO:

INICIO

DEFINIR cola COMO COLA ENTERO

ENCOLAR(cola, 3)

ENCOLAR(cola, 7)

ENCOLAR(cola, 9)

ELEMENTO \leftarrow DESENCOLAR(cola)

ESCRIBIR "Elemento eliminado: ", ELEMENTO

ESCRIBIR "Elemento al frente: ", FRENTE(cola)

FIN





Ejercicios

Ejercicios de listas

Resolver los siguientes ejercicios en pseudocódigo a lápiz y papel

Ejercicio para solucionar con listas (Utilice métodos de ser necesario):

1. Escribe un programa que solicite al usuario N números enteros y los almacene en una lista. Luego, el programa debe calcular el promedio de todos los números ingresados.
2. Escribe un programa que solicite al usuario N números enteros y los almacene en una lista. Luego, el programa debe contar cuántos números son pares y cuántos son impares, mostrando el resultado.
3. Crea un programa que permita al usuario ingresar N números y los almacene en una lista. Luego, ordena la lista en orden ascendente e imprime el resultado.
4. Crea un programa que solicite N números y los almacene en una lista. Luego, elimina los números repetidos y muestra la lista sin duplicados.
5. Escribe un programa que permita al usuario ingresar dos listas de números y luego las fusione en una sola lista ordenada.

Ejercicios de pilas

Resolver los siguientes ejercicios en pseudocódigo a lápiz y papel

Ejercicio para solucionar con pilas (Utilice métodos de ser necesario):

1. Escribe un programa que permita ingresar una serie de números y los imprima en orden inverso usando una pila
2. Simula la operación de una torre de control en un aeropuerto usando una pila.
 - Cada avión que solicite aterrizar se apila
 - Cuando un avión aterriza, se desapila
 - Debe haber una opción para mostrar la lista de aviones en espera.
3. Crea un programa que simule una pila de cartas en un juego. Las opciones que tendrá el programa serán las siguientes:
 - Agregar una carta a la pila.
 - Robar la última carta (desapilar).
 - Mostrar la pila de cartas.
 - Salir.

Ejercicios de colas

Resolver los siguientes ejercicios en pseudocódigo a lápiz y papel

Ejercicio para solucionar con colas (Utilice métodos de ser necesario):

1. Simula una cola de atención en un banco donde los clientes llegan en orden y son atendidos en el mismo orden (FIFO - First In, First Out). Las opciones que tendrá el programa son las siguientes:
 - Agregar un cliente a la cola.
 - Atender al siguiente cliente (desencolar).
 - Mostrar la cola actual.
2. Simula una cola de pacientes en un hospital donde se atienden en orden de prioridad en lugar de FIFO. Las opciones que tendrá el programa son las siguientes:
 - Agregar paciente con nivel de prioridad (1 urgente, 2 medio, 3 leve).
 - Atender al paciente con mayor prioridad.
 - Mostrar la lista de pacientes en espera.