

Pilas y Colas

Contenidos

- 1 Pila
 - El Concepto
 - Funcionamiento y operaciones básicas
 - Declaración de una Pila
 - Operaciones
 - Eficiencia de las Operaciones
 - Aplicaciones

- 2 Cola
 - El Concepto
 - Ejemplos reales
 - Funcionamiento y operaciones básicas
 - Tipos de colas
 - Operaciones

- 3 Ejercicios
 - Pila
 - Cola

Pila

El Concepto

- Una pila (*stack*) es una colección ordenada de elementos a los que sólo se puede acceder por un único lugar o extremo de la pila.
- El método de pila para la evaluación de expresiones fue propuesto en 1955 y dos años después patentado por Fiedrich L. Bauer, quién recibió en 1988 el premio "*IEEE Computer Society Pioneer Award*" por su trabajo en el desarrollo de dicha estructura de datos.
- Son una estructura de almacenamiento de información bastante usual. El concepto de pila aparece en la vida diaria: pila de libros, de monedas, de platos.

Pila

Ejemplos reales

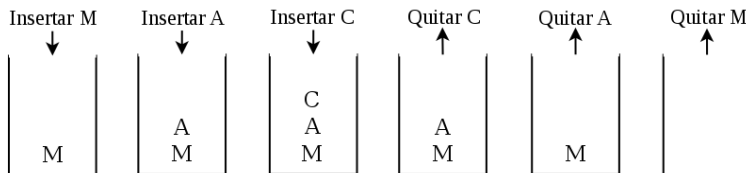


Pila

- La pila se utiliza siempre que se quiera recuperar una serie de elementos en orden inverso a como se introdujeron.
- Tanto la extracción de elementos como la inserción, se realiza por la parte superior, lo que implica que el único elemento accesible de una pila sea el último.
- Como el último elemento que se pone en la pila es el primero que se puede sacar, a estas estructuras se les conoce por el nombre de LIFO (*Last In First Out*).

Pila

Funcionamiento y operaciones básicas



Entrada: MAC

Salida: CAM

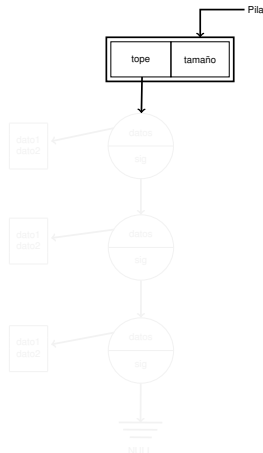
Tipo de dato	Dato que se almacena en la pila.	
Crear	Se crea la pila vacía.	<i>constructor</i>
Tamaño	Regresa el numero de elementos de la pila.	<i>size</i>
Apilar	Se añade un elemento a la pila.	<i>push</i>
Desapilar	Se elimina el elemento frontal de la pila.	<i>pop</i>
Tope	Devuelve el elemento que esta en la cima de la pila.	<i>top</i>
Vacía	Devuelve cierto si la pila está vacía o falso en caso contrario.	<i>empty</i>
Llena	Devuelve cierto si la pila está llena o falso en caso contrario.	<i>full</i>

Pila

Declaración de una Pila: La Pila

- Una pila vacía no contiene nodos.
- Se debe conocer solo el tope.
- Se debe conocer el tamaño actual de la pila

```
1 typedef struct pila {  
2     Nodo *tope;  
3     int tamano;  
4 } Pila;
```

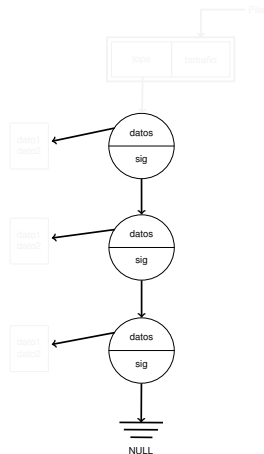


Pila

Declaración de una Pila: El nodo

- Cada nodo debe almacenar información.
- Cada nodo debe almacenar la dirección del nodo siguiente.

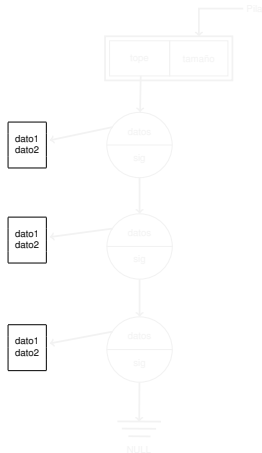
```
1 typedef struct nodo {  
2     Info *datos;  
3     struct nodo *siguiente;  
4 } Nodo;
```



Pila

Declaración de una Pila: La información

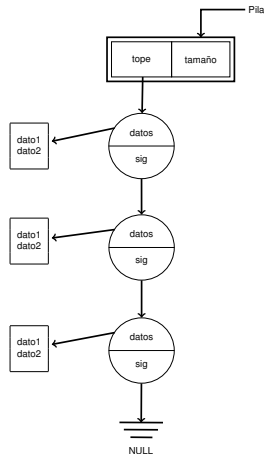
```
1 typedef struct info {  
2     int dato1;  
3     /* int dato2; */  
4 }Info;
```



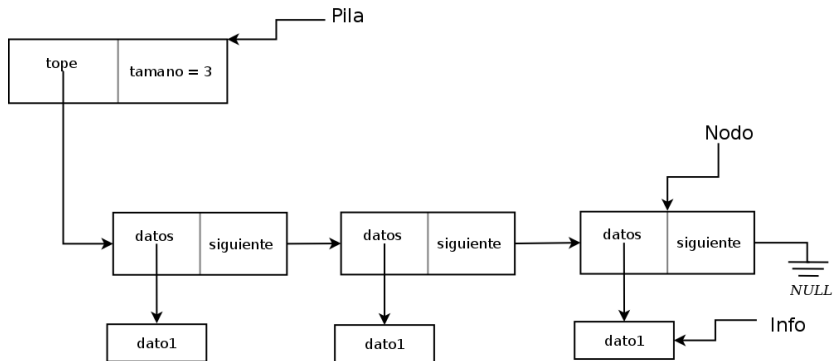
Pila

Declaración de una Pila

```
1 typedef struct info {  
2     char dato1;  
3     /* int dato2; */  
4 }Info;  
5  
6 typedef struct nodo {  
7     Info *datos;  
8     struct nodo *siguiente;  
9 } Nodo;  
10  
11 typedef struct pila {  
12     Nodo *tope;  
13     int tamano;  
14 } Pila;
```



Pila



Pila

Operaciones: Crear una pila

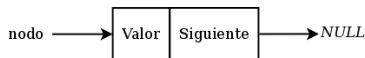
- Crear una pila: hacer que tope apunte a `NULL`.

`top` \longrightarrow `NULL`.

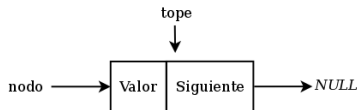
Pila

Operaciones: Insertar en una Pila vacía

- 1 Crear un nodo y hacer que su siguiente apunte a **NULL**.



- 2 Hacer que tope apunte a nodo.



Pila

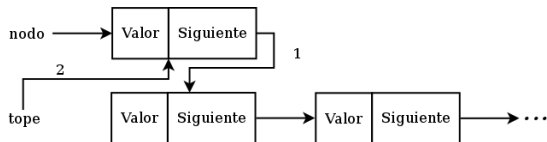
Operaciones: Insertar en una Pila no vacía

- 1 Crear un nodo y hacer que su siguiente apunte al tope.
- 2 Hacer que tope apunte al nodo.

- Antes



- Después



Pila

Operaciones: Leer y eliminar un elemento

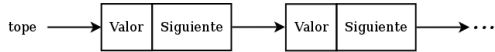
Suponiendo que se parte de una pila con uno o más nodos, considere un apuntador auxiliar nodo:

- 1 Hacer que nodo apunte al primer elemento de la pila, es decir a tope.
- 2 Asignar a tope la dirección del segundo nodo de la pila; es decir, el de su nodo siguiente.
- 3 Guardar el contenido de nodo para devolverlo como retorno.
- 4 Liberar la memoria asignada a nodo, que es el que se desea eliminar.

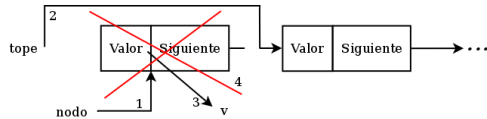
Pila

Operaciones

- Antes



- Después



Pila

Eficiencia de las Operaciones

Operaciones	Tiempo
Tamaño	$O(1)$
Vacía	$O(1)$
Tope	$O(1)$
Apilar	$O(1)$
Desapilar	$O(1)$

Pila

Aplicaciones

- Llamada a subprogramas.
- Eliminación de recursividad.
- Tratamiento de expresiones aritméticas.
- Pilas a nivel de arquitectura hardware es la asignación de memoria.
- Reconocedores sintácticos de lenguajes independientes del contexto.
- Solucionar problemas de búsqueda.

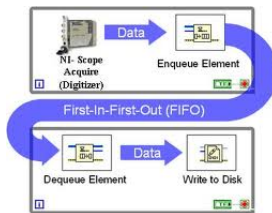
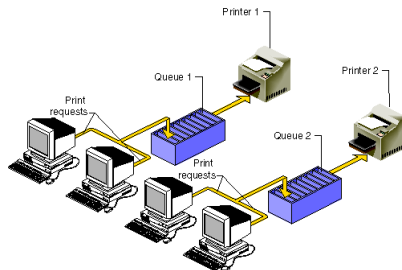
Cola

El Concepto

- Una cola (*queue*) es una colección de elementos, donde estos se introducen por un extremo y se eliminan por el otro.
- A estas estructuras se les conoce FIFO (*First In First Out*), debido a que el primer elemento en entrar será también el primero en salir.
- Las colas se utilizan en la ciencias de la computación, transportes y operaciones de investigación (entre otros), dónde los objetos, personas o eventos son tomados como datos que se almacenan y se guardan mediante colas para su posterior procesamiento.

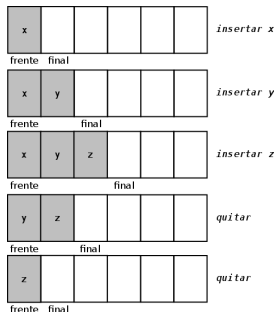
Cola

Ejemplos reales



Cola

Funcionamiento y operaciones básicas



Tipo de dato	Dato que se almacena en la cola.	
Crear	Se crea la cola vacía.	constructor
Tamaño	Regresa el numero de elementos de la cola.	<i>size</i>
Encola	Se añade un elemento a la cola.	<i>push</i>
Desencola	Se elimina un elemento de la cola.	<i>pop</i>
Comienzo	Devuelve el elemento que está al comienzo de la cola.	<i>front</i>
Final	Devuelve el elemento que está al final de la cola.	<i>last</i>
Vacía	Devuelve cierto si la cola está vacía o falso en caso contrario.	<i>empty</i>
Llena	Devuelve cierto si la cola está llena o falso en caso contrario.	<i>full</i>

Cola

Tipos de colas

- Colas Circulares.
- Bicolos o doble colas.
- Colas con prioridad

Cola

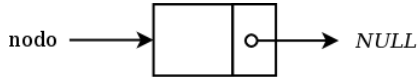
Operaciones: Crear Cola

- Crear una Cola: hacer que *inicio* y *fin* apunte a **NULL**. $inicio \rightarrow \text{NULL}$
 $fin \rightarrow \text{NULL}$

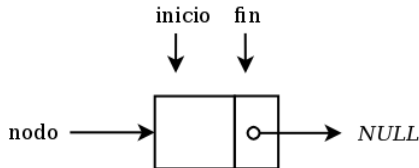
Cola

Operaciones: Insertar en una Cola vacía

- 1 Crear un nodo y hacer que su siguiente apunte a **NULL**.



- 2 Hacer que *inicio* y *fin* apunte a nodo.

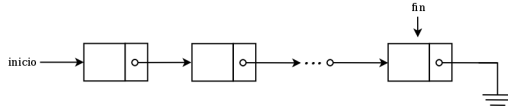


Cola

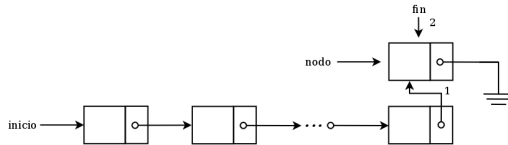
Operaciones: Insertar en una Cola no vacía

- 1 Crear un nodo y hacer que el siguiente del último nodo apunte al nuevo nodo.
- 2 Hacer que *fin* apunte al nuevo nodo.

- Antes



- Después



Cola

Operaciones: Eliminar un elemento

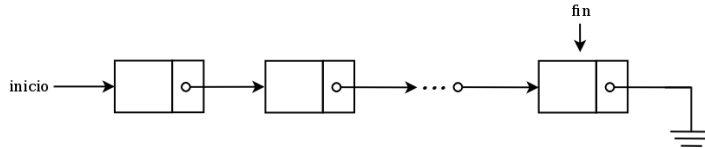
Suponiendo que se parte de una cola con uno o más nodos, considere un apuntador auxiliar nodo:

- 1 Hacer que nodo apunte al primer elemento de la cola, es decir a *inicio*.
- 2 Asignar a *inicio* la dirección del segundo nodo de la cola; es decir, el de su nodo siguiente.
- 3 Liberar la memoria asignada a nodo, que es el que se desea eliminar.

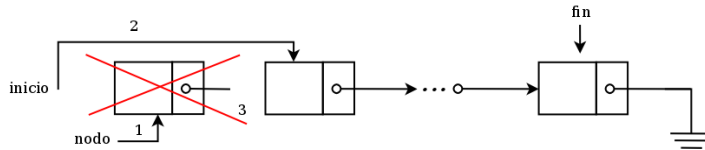
Cola

Operaciones: Eliminar un elemento

- Antes



- Después



Ejercicios

Pila

- 1 Verifique si una palabra es palíndroma.

Ejercicios

Cola

- ❶ **El problema de Josefo:** Se tienen n personas entorno a un círculo, ordenadas y numeradas desde la primera a la n -ésima. Empezando por la persona número 1, se saltan $m - 1$ personas y se mata a la m -ésima. A continuación se saltan otras $m - 1$ personas y se ejecuta a la siguiente. El proceso continúa hasta que sólo quede una. El objetivo es encontrar el lugar inicial en el círculo para sobrevivir $J_m(n)$, dados n y m .

Ejemplo $n = 12$ y $m = 3$, el sobreviviente es 10.

Pilas y Colas