

TAD Pila

ALGORITMOS Y
ESTRUCTURAS DE
DATOS I

Pila (Stack)

- ❑ Las pilas son listas con la restricción que las inserciones y eliminaciones debe realizarse solamente en una posición: la cima de la pila (final de la lista), llamada **top**.
- ❑ Las pilas son conocidas también como LIFO (Last In First Out, último en ingresar es el primero en irse)

(Solo el elemento top es accesible)



Pila (Stack)

Operaciones básicas

push

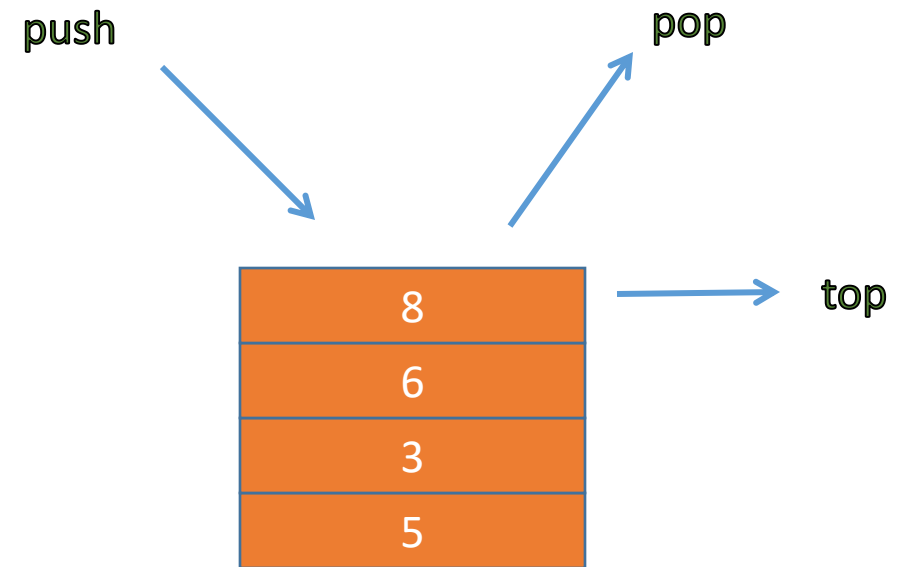
Insertar un elemento
(la pila no está llena)

pop

Elimina el ultimo elemento insertado
(la pila no está vacía)

top

Retorna el ultimo elemento insertado
(la pila no está vacía)



Pila (Stack)

Ejemplo

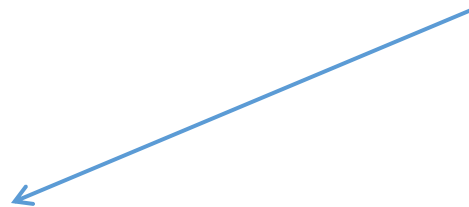
Si realizamos las siguientes operaciones obtendremos:

```
unaPila.Push(5)  
unaPila.Push(3)  
unaPila.Push(6)  
unaPila.Push(8)
```

```
unaPila.pop()  
unaPila.pop()
```

```
unaPila.top()
```

3



8
6
3
5

Pila (Stack)

Operaciones básicas (implementación con lista y tope)

//Inserta al inicio de la lista (tope)

public void apilar(int dato);

//Elimina el primer elemento de la lista (pop)

public void desapilar();

//Indica si la pila esta vacía

public boolean esVacia();

//Indica si la pila esta llena

public boolean esllena();

//Retorna el nodo al tope de la lista (top)

public Nodo cima();

//Retorna la cantidad de elementos de la pila

public int elementos();

Pila (Stack)

Implementación: Lista enlazada



Array



Pila: Lista vs Array

Lista Enlazada: Estructura dinámica. Todas las operaciones requieren de tiempo constante para su ejecución, ya que no dependen del tamaño (excepto la eliminación de toda la estructura).

Arrays: Estructura estática. Todas las operaciones requieren de tiempo constante para su ejecución, ya que no dependen del tamaño. Evita el uso de punteros. Es la solución más popular. Requiere conocer el tamaño de la pila.