# Pilas

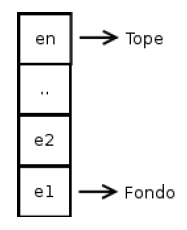
Una pila (stack en inglés) es una lista ordinal o estructura de datos en la que el modo de acceso a sus elementos es de tipo LIFO (del inglés Last In First Out, último en entrar, primero en salir) que permite almacenar y recuperar datos. Se aplica en multitud de ocasiones en informática debido a su simplicidad y ordenación implícita en la propia estructura.

Imagen que contiene captura de pantalla

Descripción generada con confianza alta

Las pilas son muy utilizadas en programación, para evaluar expresiones, reconocer lenguajes, recorrer árboles y simular procesos recursivos. En todo momento, el único elemento visible de la estructura es el último que se colocó. Se define el tope de la pila como el punto donde se encuentra dicho elemento, y el fondo, como el punto donde se encuentra el primer elemento incluido en la estructura.

La administración de una pila se puede hacer con muy pocas operaciones: una constructora (permite crear pilas vacías), dos modificadoras (para agregar y eliminar elementos) y dos analizadoras (retornar el elemento del tope, e informar si la pila está vacía). Se incluye, además, una destructora para retornar el espacio ocupado por la pila. Por simplicidad, no se contemplan operaciones de persistencia. El formalismo escogido para referirse al objeto abstracto pila se muestra a continuación. Allí se da un nombre a cada uno de los elementos que hacen parte de la estructura y se marca claramente el tope y el fondo.



Si la pila no tiene ningún elemento se dice que se encuentra vacía y no tiene sentido referirse a su tope ni a su fondo. Una pila vacía se representa con el símbolo Ø. Por último, se define la longitud de una pila como el número de elementos que la conforman.

Una pila tiene las siguientes operaciones básicas:

* Crear: se crea la pila vacía
* Apilar: (push), que coloca un elemento en la pila
* Retirar (o desapilar, pop), es la operación inversa que retira el último elemento apilado
* Cima: devuelve el elemento que está en la cima de la pila (top o peek).
* Vacía: devuelve true si la pila está vacía o falso en caso contrario.

# Bibliografías

1. Brassard, G., y Bratley, P.: Fundamentos de algoritmia Madrid. Prentice-Hall, 1997.
2. Heileman, Gregory L.: Estructuras de datos, algoritmos, y programación orientada a objetos, Madrid, McGraw-Hill, 1997.
3. Estructura de datos: Algoritmos, abstracción y objetos, Luis Joyanes, McGraw-Hill, España, 1998.
4. Ingeniería Informática: Procesadores de lenguaje, universitat Jaume.
5. Franch Gutierrez, Estructura de datos, especificación, diseño e implementación, Barcelona, Ediciones UPC, 1994.
6. Collins, William J, Data Structures, An object-Oriented Approach, Addison-Wesley, 1992.