

# Algoritmos y Estructuras de Datos

## Práctica 2: Pilas y cadenas.

Marco Antonio Heredia Velasco

**Fecha de entrega: Viernes 6 de junio (hasta media noche).**

### 1. Instrucciones de entrega

Las instrucciones de entrega son las mismas de la práctica anterior: Se deberá realizar en equipos de a lo más 3 personas. Todos los archivos deben estar contenidos en un directorio cuyo nombre estará formado por el primer apellido de los integrantes del equipo. Dentro debe haber 2 subdirectorios: **docs** y **fuentes**. Se debe empaquetar esta estructura de directorios en un archivo comprimido, llamado igual que el directorio principal, pero con extensión determinada por el compresor utilizado (Ej. **HerediaUrrutiaPerez.zip**). Enviar dicho archivo al correo **hvma@correo.azc.uam.mx** con asunto: *Práctica 2 AED*.

Las prácticas que no cumplan con estas instrucciones, no serán revisadas. La práctica puede ser elaborada en C, C++ o Java.

### 2. Programa

Como se vió en clase, una Pila es una estructura de datos lineal de cero o más celdas que contienen el mismo tipo de dato. Se caracteriza porque el último elemento que ingresa a ella es el primero en salir (LIFO) y se manipula con las siguientes operaciones (ver Figura 1):

**push(*elem*)** Coloca el nuevo elemento *elem* en el tope de la pila.

**pop()** Elimina la celda que se encuentra en el tope de la pila y actualiza el tope (no regresa nada).

**tope()** Regresa el elemento que se encuentra en el tope de la pila sin quitarlo de ella.

`es_vacia()` Determina si una pila tiene elementos o no (pila vacía).

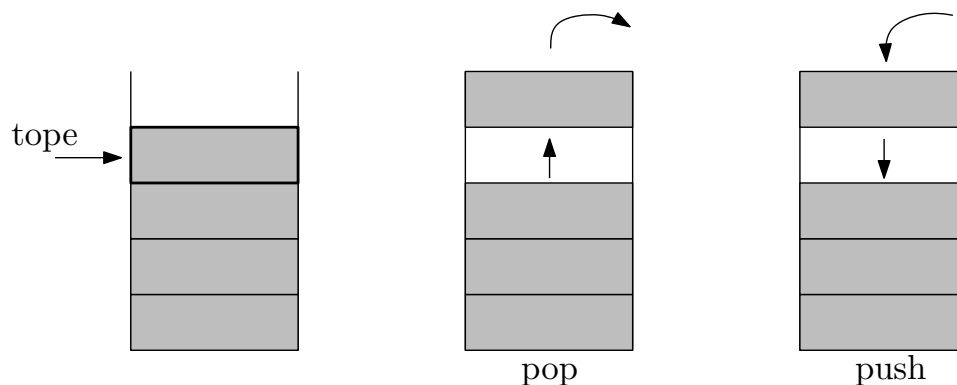


Figura 1: Ejemplo de operaciones en pila.

## 2.1. Actividades

Implementar la estructura de datos Pila de *caracteres* y sus métodos asociados.

Utilizando la estructura anterior, implemente un programa llamado *reversa* que dada una cadena entregue su cadena reversa, por ejemplo, la reversa de la cadena "Hola Ana." es la cadena ".anA aloH".

## 2.2. Especificaciones

El código fuente de su programa deberá estar en el directorio **fuentes** y debe estar ampliamente comentado. Sólo deben mandar los archivos con el código fuente, no deben incluir ningún ejecutable.

Su implementación de Pila deberá tener las operaciones sobre pilas descritas arriba, más otra operación que inicialice la estructura de pila para poder empezarla a utilizar (pila vacía).

El programa *reversa* solicitará que el usuario introduzca con el teclado una oración de a lo más 20 caracteres. Se procesará esa oración utilizando una Pila y se imprimirá en pantalla (salida estándar) la cadena reversa de la oración recibida. Deben poner especial cuidado con el manejo del carácter `'\0'` de fin de cadena.

Se deberá implementar la Pila (y sus métodos) de forma independiente del programa que calcula la reversa, o sea, el programa que calcula la reversa debe importar al de la Pila para poder usarlas.

### 3. Documentos

El directorio `docs` deberá contener un archivo `README.txt` en texto plano indicando el nombre completo de los integrantes del equipo, el lenguaje de programación utilizado, el sistema operativo en que probaste tu programa, y la forma adecuada de compilar y correr su programa.