



# Universidad Autónoma de Zacatecas

Unidad Académica de Ingeniería Eléctrica

Programa Académico de Ingeniería de Software

## Práctica 9

### Datos generales:

Nombre de la Práctica	Pilas
Nombre de la carrera	Ingeniería de Software
Nombre de la materia	Laboratorio de Estructuras de Datos
Número y nombre de Unidad(es) temática(s)	I1. Estructuras lineales.
Docente que imparte la materia	Aldonso Becerra Sánchez
Fecha de entrega para los alumnos	4-abril-2025
Fecha de entrega con extensión y penalización	6-abril-2025
Fecha de elaboración	3-abril-2025

Objetivo de la Práctica	Comprender el uso de las pilas en un ejemplo común.
Tiempo aproximado de realización	4 horas
Introducción	Las pilas son estructuras de datos fáciles de entender, sin embargo, sus aplicaciones muchas veces no son tan sencillas de visualizaren la práctica. Las pilas facilitan muchos procesos, un caso común son el manejo de notaciones infija, prefija y postfija.

### Referencias que debe consultar el alumno (si se requieren):

#### Referencia 1:

1.Cairo, Osvaldo; Guardati, Silvia. Estructura de Datos, Tercera Edición. McGraw-Hill, México, Tercera Edición, 2006.

#### Referencia 2:

2.Mark Allen Weiss. Estructura de datos en Java. Ed. Addison Wesley.

#### Referencia 3:

3. Joyanes Aguilar, Luis. Fundamentos de Programación. Algoritmos y Estructuras de Datos. Tercera Edición, 2003. McGraw – Hill.



# Universidad Autónoma de Zacatecas

Unidad Académica de Ingeniería Eléctrica

Programa Académico de Ingeniería de Software

---

## Actividades que debe realizar el alumno:

### Actividad inicial:

Generar el reporte en formato IDC.

### Actividad 1:

Primero genere la **introducción**.

### Actividad 2:

Hacer un programa que funcione como una calculadora.

Este programa permitirá funcionar como una calculadora, debe recibir una expresión aritmética en una caja de texto (o el teclado), por ejemplo de la forma:

$(a-3.6)*f/2^{(c+1)}$

Asuma que la entrada es una expresión válida, es decir no deberá realizar validaciones de si está bien formada la expresión (eso lo deberá hacer otro módulo en otro programa, que ahorita no se está pidiendo; que pudiera usarse independientemente del funcionamiento del programa que se le pide a usted en esta actividad práctica).

El programa deberá recibir de entrada esta cadena con la expresión aritmética. Posteriormente se deberá obtener el resultado de la operación ejecutada (evaluar la expresión).

Consideraciones de diseño:

- Considere variables de un solo carácter.
- Revise la expresión aritmética para checar si tiene variables (de una sola letra); en caso de que tenga, pida los valores que esas variables debe tener para sustituirlas por su contenido.
- Considere que los números son enteros y decimales, y de cualquier longitud. Ejemplos: 456, 56, 4, 44443, 4.5, 9.99.
- Al final el programa deberá obtener el resultado de la expresión infija ingresada, para ello deberá convertir la cadena en una expresión “postfija” o “prefija” (usando pilas), para posteriormente realizar la evaluación de dicha expresión convertida (usando pilas).

Ejemplo:

**Entrada:**  $(a-2.0)*f/2^{(c-8)}$

donde  $a=1$ ,  $f=3$ ,  $c=12.0$

**Salida:** -0.1875



# Universidad Autónoma de Zacatecas

Unidad Académica de Ingeniería Eléctrica

Programa Académico de Ingeniería de Software

---

NOTA: recuerde que no puede usar librerías de Java que automaticen ningún proceso de manipulación de grupos de datos, en este caso de cadenas. Por decir, están prohibidos los usos de expresiones regulares.

Haga el programa (actividad 2, la cual es el **Desarrollo** del programa, junto con la captura de pantalla del programa funcionando).

## Actividad 3:

Una manera sencilla de encriptar mensajes, es colocar paréntesis de manera arbitraria, y todo lo que está dentro de un paréntesis (ponerlo al revés el contenido), por ejemplo “Curso de Informática” se puede encriptar como “Cur(os) de I(rofn)mática”.

1. Escriba un programa, que dado un mensaje en texto plano “Curso de Informática”, encripte el mensaje pidiendo las posiciones a usar de los paréntesis, sabiendo que pueden usarse todos los pares paréntesis que se deseen; así mismo debe pedirse el tamaño de la agrupación de los paréntesis. Por ejemplo: usar 3 agrupaciones, y usarse en posición 4 y 11 (de la cadena original), indicando que el tamaño del primer paréntesis es 2 (“os”), el tamaño del segundo es 4 (“rofn”), y del tercero es 3 (“cit”); dando como resultado “Cur(os) de I(rofn)má(cit)a”. Las restricciones a usar solo son que las posiciones no deben empalmarse. Esto se logra validando las posiciones y los tamaños con respecto a la longitud de la cadena. Si no se cumplen las reglas, el programa no debe poder encriptar.
2. Escriba un programa que permita des-encriptar siguiendo las mismas reglas que el punto 1.

## Actividad 4:

Pruebe el funcionamiento del programa de la actividad 2 con todo y sus capturas de pantalla.

## Actividad 5:

Realice la sección de **Código agregado** (diagrama de clases UML).

## Actividad 6:

Realice la sección de **Pre-evaluación** (use los lineamientos establecidos).



# Universidad Autónoma de Zacatecas

Unidad Académica de Ingeniería Eléctrica

Programa Académico de Ingeniería de Software

---

## Actividad 7:

Finalmente haga las **Conclusiones**.

## Actividad 8:

Enviar en <http://ingsoftware.reduaz.mx/moodle>

Archivo anexo que se requiere para esta tarea (opcional):

Dudas o comentarios: [a7donso@gmail.com](mailto:a7donso@gmail.com)