

Estructura de Datos	TRABAJO PRÁCTICO N° 3 - Ciclo 2025 Tema: Pila (Stack)
INGENIERÍA INFORMÁTICA – LICENCIATURA EN SISTEMAS FACULTAD DE INGENIERÍA – UNIVERSIDAD NACIONAL DE JUJUY	

El trabajo práctico permite conocer, distinguir, desarrollar y utilizar las estructuras de datos que implementan el tipo de dato abstracto Stack o Pila.

Este trabajo también está pensado para que cada estudiante desarrolle habilidades de búsqueda y selección de información técnica y/o científica necesaria para resolver los ejercicios.

Para la resolución se debe usar la implementación de Stack propuesta en clases.

Ejercicios propuestos

- 1) Escribir un programa que genere 20 números aleatorios y los guarde en una pila. Luego, quitar los elementos de la pila y guardar los números pares en una nueva pila y los impares en otra. Calcular y mostrar el promedio y la suma total de los números que poseen cada una de las pilas creadas. Al finalizar el proceso, la pila original debe mantenerse sin modificaciones.

Indicaciones:

Este ejercicio necesita del objeto scanner para ingresar datos por la consola o teclado, se espera que el código **controle los problemas** que normalmente ocurren al operar con la consola o teclado.

Se espera una **correcta modularización** entre el código que realiza el **ingreso** y **validación** de los datos respecto del código que hace lo que se **solicita en el ejercicio**. También necesita del objeto random para generar valores de manera aleatoria.

- 2) Dado un arreglo de números enteros (positivos, negativos y ceros), se desea invertir únicamente los números negativos. Para la inversión se debe utilizar una pila y se debe mantener el resto del arreglo sin modificaciones.

Ejemplo:

Arreglo original:

4	-1	7	-3	-5	2	9	15	-8
---	----	---	----	----	---	---	----	----

Números negativos: -1, -3, -5, -8, se invierten usando una pila, -8, -5, -3, -1

Resultado:

4	-8	7	-5	-3	2	9	15	-1
---	----	---	----	----	---	---	----	----

Indicaciones:

Este ejercicio necesita del objeto scanner para ingresar datos por la consola o teclado, se espera que el código **controle los problemas** que normalmente ocurren al operar con la consola o teclado.

Estructura de Datos	TRABAJO PRÁCTICO N° 3 - Ciclo 2025 Tema: Pila (Stack)
INGENIERÍA INFORMÁTICA – LICENCIATURA EN SISTEMAS FACULTAD DE INGENIERÍA – UNIVERSIDAD NACIONAL DE JUJUY	

Se espera una **correcta modularización** entre el código que realiza el **ingreso** y **validación** de los datos respecto del código que hace lo que se **solicita en el ejercicio**.

- 3) Dada una pila cargada con N números enteros generados de manera aleatoria, implementar lo siguiente:
 - a) Un método que reciba como parámetro una pila de enteros y dos números. El método debe eliminar los elementos menores al primer parámetro y los mayores al segundo parámetro. Los demás elementos deben mantenerse en la pila en el mismo orden.
 - b) Un método que duplique los números pares que se encuentren en la pila manteniéndose los demás sin alteraciones.
 - c) Un método que intercambie el primer elemento (el de la base) con el último elemento de la pila (el de la cima).
 - d) Un método que compare la suma de la mitad inferior y la mitad superior de la pila devolviendo cual mitad es la mayor. La pila no debe modificarse.

Indicaciones:

Este ejercicio necesita del objeto scanner para ingresar datos por la consola o teclado, se espera que el código **controle los problemas** que normalmente ocurren al operar con la consola o teclado.

Se espera una **correcta modularización** entre el código que realiza el **ingreso** y **validación** de los datos respecto del código que hace lo que se **solicita en el ejercicio**. También necesita del objeto random para generar valores de manera aleatoria.

- 4) Implementar la clase Evento que posea los siguientes atributos: tipo (info, advertencia, error), descripción y código. Utilizar una pila para almacenar los eventos recientes del sistema. El programa debe implementar los siguientes métodos:
 - a) registrarEvento: recibe como parámetro la pila y el evento, debe agregar el evento a la pila.
 - b) eliminarEventosDeTipo: recibe como parámetro la pila y el tipo, elimina todos los eventos del tipo especificado (por ejemplo "info"). El resto de los elementos de la pila no debe alterarse.
 - c) buscarEventoPorCodigo: recibe como parámetro la pila y un código, devuelve el primer evento que coincida con el código (sin modificar la pila).
 - d) contarErrores: recibe como parámetro la pila y devuelve cuántos eventos son del tipo "error".
- 5) Dada una pila de caracteres, implementar una compresión en la cual secuencias de letras consecutivas iguales son convertidas a una letra seguida de su cantidad.
 Por ejemplo, una pila que contiene 'a', 'a', 'a', 'b', 'b', 'c' se convierte en 'a3', 'b2', 'c1'.
 El programa debe implementar los siguientes métodos:

Estructura de Datos	TRABAJO PRÁCTICO N° 3 - Ciclo 2025 Tema: Pila (Stack)
INGENIERÍA INFORMÁTICA – LICENCIATURA EN SISTEMAS FACULTAD DE INGENIERÍA – UNIVERSIDAD NACIONAL DE JUJUY	

- a) generarEtiquetas(Stack <Character> pila) Devuelve una pila de String llamada etiquetas agrupando los caracteres consecutivos iguales y contando su frecuencia, por ejemplo "a3", "b2", ...".
- b) buscarEtiqueta(Stack <String> etiquetas, char letra) Retorna la etiqueta correspondiente a una letra dada. Si no la encuentra devuelve null.
- c) expandirEtiqueta(Stack <String> etiquetas) Reconstruye la pila original a partir de la pila de etiquetas, por ejemplo, si la pila de etiquetas tiene "c4", la pila debe contener 'c', 'c', 'c', 'c'.

Indicaciones:

Este ejercicio necesita del objeto scanner para ingresar datos por la consola o teclado, se espera que el código **controle los problemas** que normalmente ocurren al operar con la consola o teclado.

Se espera una **correcta modularización** entre el código que realiza el **ingreso y validación** de los datos respecto del código que hace lo que se **solicita en el ejercicio**.

- 6) Definir la clase Entrega con los atributos destinatario, piso y fueEntregado. Cargar una pila con varios objetos de tipo Entrega. Además, implementar los siguientes métodos:
 - a) cargarEntrega(Stack <Entrega> pila, Entrega nueva): agrega una nueva entrega a la pila.
 - b) marcarComoEntregado(Stack <Entrega> pila, String destinatario): cambia el valor de fueEntregado a true para ese destinatario.
 - c) obtenerPendientesPorPiso(Stack <Entrega> pila, int piso): devuelve cuántas entregas pendientes hay en ese piso.
 - d) vaciarEntregados(Stack <Entrega> pila): elimina todos los objetos que ya fueron entregados.

Indicaciones:

Este ejercicio necesita del objeto scanner para ingresar datos por la consola o teclado, se espera que el código **controle los problemas** que normalmente ocurren al operar con la consola o teclado.

Se espera una **correcta modularización** entre el código que realiza el **ingreso y validación** de los datos respecto del código que hace lo que se **solicita en el ejercicio**.