



# Manual de prácticas

ADT lineales

Ejercicios aplicados para el uso TDA lineales

Este manual contiene actividades prácticas para aplicar los TDA lineales como Pilas,  
Colas, Listas simples, dobles y circulares

Zaira Ruth Zuviría, TIME \@ "dd/MM/yyyy" 09/02/2023  
zzuviria@edu.uag.mx

## Contenido

Práctica No. 1 – Verificador de pares	2
Práctica No. 2 – servidor de impresión	3
Práctica No. 3 – Pilas y colas	4
Práctica No. 4 – Memoria dinámica	5
Práctica No. 5 – Lista Ordenada	6



## Práctica No. 1 - Verificador de pares

---

**Objetivo:** Aplicar la funcionalidad del TDA Pila para validar el acomodo de paréntesis anidados.

### Instrucciones:

Las pilas son estructuras de datos que permite el almacenamiento de elementos utilizando el algoritmo LIFO (Last In, First Out), las operaciones que debe tener implementadas son push() -insertar- y pop() – eliminar-. Utilice este TDA para verificar si en una cadena tiene acomodados adecuadamente los pares de paréntesis, corchetes y llaves. Por ejemplo:  $(a+b)^* \text{ vel+ } [\text{tiempo} / \{\text{aceleracion}\}]$

La fórmula no es importante pero el acomodo de los pares de paréntesis sí. Cada vez que encuentre un símbolo de apertura al recorrer la cadena, envíelo a la pila (push()), cada vez que localice uno de cerradura haga un pop() y valide que “hagan par”. Si no son pares el programa termina. Si todo va correctamente revise hasta llegar al final de la cadena.

### Puntuación

1. El programa analiza una cadena con pares de paréntesis en cualquier orden (15 pts)
2. Utiliza las funciones push() y pop() para acceder a la pila (25 pts)
3. Notifica de un fallo en el acomodo de los símbolos (),[],{} en la cadena (25 pts)
4. Notifica que la cadena está correctamente anidada (25 pts)
5. Entrega el programa en extensión .c, con comentarios, buena indentación y datos del programa y equipo (10 pts)



## Práctica No. 2 - servidor de impresión

---

**Objetivo:** Aplicar la funcionalidad del TDA Cola para ejemplificar una aplicación en la vida real.

### Instrucciones:

Las colas son Tipos de dato abstracto que funcionan con el algoritmo FIFO (First In – First out), es de uso común en establecimientos donde importa el orden de llegada para atender a los usuarios. Las colas de impresión funcionan de la misma manera, acomodando las peticiones de los trabajos conforme fueron llegando para seguir el mismo orden de salida si no se ha contemplado alguna prioridad, por lo que, en esta práctica se utilizarán las funciones insertar() y eliminar() para acceder al TDA y aplicarlo en un programa que permita a través de un menú que permita enviar a impresión, imprimir todos los trabajos y salir.

Para enviar una petición es necesario indicar el nombre de usuario para llevar el registro y validar el acceso. Una vez ingresado el nombre de usuario, se debe solicitar el carácter a imprimir y el número de apariciones:

Usuario “ABC”, ‘@’, 20

Usuario “XYZ”, ‘\$’, 10

La salida debería ser:

“ABC”

@@@@@@@@@@@@@@@@@@

“XYZ”

\$\$\$\$\$\$\$\$\$\$

### Puntuación

1. El programa tiene un menú de opciones (10 pts)
2. Permite enviar varios trabajos a impresión (15 pts)
3. La opción imprimir muestra los trabajos en orden FIFO (20 pts)
4. Las operaciones utilizan insertar - in() (20 pts) y eliminar - out() (20 pts) para acceder a la cola
5. Entrega el código en .c con datos generales del programa y propios, bien indentados (10 pts)
6. Su envío es en tiempo y forma (5 pts)



## Práctica No. 3 - Pilas y colas

---

**Objetivo:** Repasar los algoritmos relacionados con los TDA Pilas y Colas para resolver problemas cotidianos.

**Instrucciones:**

Justifique el uso del TDA apropiado para resolver cada uno de los problemas presentados

- a) Una escuela mantiene un programa de incentivos para los profesores que llegan puntualmente a lo largo de la semana. Implemente un control que permita conocer el orden de llegada de los profesores y mantener en un listado a los que llegaron después de la hora de entrada.
- b) Los estudiantes de universidad desarrollaron una calculadora de operaciones simples, para ejecutarlas convierten a notación post-order, es decir dos operadores y un operador:  $3+5*8$  se transforma en  $3\ 5\ +\ 8\ *$  ejecutándose  $3\ 5\ +\ \rightarrow 8\ * \ 8 \rightarrow 64$  (resultado)
- c) Una refresquera tiene los datos sobre los refrescos más vendidos. Los proveedores capturan la información y se acomodan conforme la captura. Durante el análisis se considerarán en promoción los primeros 5 con ventas arriba de 8000 unidades. Muestre cuáles marcas tendrán el beneficio y qué proveedor hizo la captura.
- d) Un invernadero almacena costales de fertilizante en acomodos de 5k y 10k. Tiene 20 costales en cada montón. Ayude con un programa para simular las ventas y “observar” cómo se modifican los montones de costales al paso de los días.

### Puntuación

1. Elección del TDA (5 pts) por cada uno. (20 en total)
2. Implementación (15 pts) por cada uno. (60 en total)
3. Resuelve los 4 problemas (5 pts)
4. Entrega los programas independientes, bien acomodados y documentados (10 pts)
5. Entrega en tiempo y forma (5 pts)



## Práctica No. 4 - Memoria dinámica

---

**Objetivo:** Solicitar memoria en tiempo de ejecución para crear arreglos reubicables.

**Instrucciones:**

Describe la funcionalidad de:

- malloc(), calloc(), realloc(), describa cómo se utilizan y las diferencias entre ellas – será mostrado en el inciso C -
- Elabore un programa que pregunte cuántas calificaciones desea capturar, solicite la memoria necesaria y almacene los datos. Muestre al final el promedio.
- Pida una memoria de tamaño 400 bytes, en los primeros 100 escriba la funcionalidad de malloc() en los siguientes 100 la de calloc() y en los penúltimos la de realloc(). Muestre un menú que elija cuál información desplegar y muestre a voluntad cada descripción. Al finalizar despliegue las diferencias entre ellos

**Puntuación**

- Programa 1
  - El programa pide el tamaño a reservar y captura los datos exactos (10 pts)
  - El programa muestra el promedio de los datos capturados (10 pts)
- Programa 2
  - Solicita 400 bytes y valida su reserva (10 pts)
  - Se revisa que la función devuelva una dirección válida (10 pts)
  - Se almacena texto en cada dirección indicada (30 pts)
  - Muestra un menú iterativo para elegir el texto a consultar(20 pts)
  - Al terminar el programa muestra la diferencia entre las funciones (10 pts)



## Práctica No. 5 - Lista Ordenada

---

**Objetivo:** Realizar las operaciones de inserción y recorrido en una lista doblemente enlazada.

### Instrucciones:

Cree un programa que por medio de un menú permita la captura de datos de un elemento para insertar a una lista dinámica doblemente enlazada. Los datos deben de quedar ordenados por lo que debe contar con funciones: una para el menú, una función para buscar el lugar a insertar y la función insertar. Considere que Insertar puede ser antes del inicio o después y desplegar. Después de insertar despliegue los datos ordenados.

### Puntuación

- El programa permite capturar datos por medio de un menú (30 pts)
- El programa despliega los datos ordenados después de ingresarlos (10 pts)
- Tiene al menos las funciones indicadas (menú, buscar, insertar, desplegar) (15 pts c/u – 60 pts en total)



Agregue algún comentario para mejorar las prácticas

=)

---

