

GUÍA DE TRABAJOS PRÁCTICOS

CURSADA 2024 – 1ER CUATRIMESTRE

Práctica IV: Colas

En cada caso se debe implementar el TAD descrito y una aplicación que, utilizando el tipo abstracto de dato, permita al usuario el ingreso de una o más instancias (según sea necesario), la aplicación de cualquiera de las operaciones y luego la obtención del resultado.

1. Construir la TAD de colas correspondientes a las implementaciones de “Arreglos” y “Punteros” teniendo en cuenta:
 - a. Definir las estructuras de datos en función a las especificaciones dadas en clase y documentadas en los PDF que se encuentran en el aula virtual de la asignatura.
 - b. Se las debe implementar y respetar estrictamente los nombres y parámetros de cada una de las operaciones del TAD especificadas.
 - c. Se las debe probar y testear de forma tal que se pueda asegurar que la TAD funciona correctamente.

Importante !!!

Los siguientes ejercicios siguientes deben se implementados y resueltos en forma genérica, esto significa que se debería poder referenciar cualquiera de las implementaciones de lista (arreglo, Puntero o cursor) y los mismos deben seguir en funcionamiento sin problemas.

2. Resolver los siguientes puntos:
 - a. Informar si un elemento dado se encuentra en la cola.
 - b. Agregar un nuevo elemento en una posición dada (colarse).
 - c. Dado un elemento sacarlo de la cola todas las veces que aparezca.
 - d. Contar los elementos de la cola.
 - e. Realizar una función que realice una copia de una cola.
 - f. Invertir el contenido de una cola sin destruir la cola original.
3. Dadas dos colas, determinar si sus contenidos son iguales tanto en posición como en datos (solo comparar por la clave), sin destruirlas. Utilizar para la resolución del problema una sola cola auxiliar. Determinar la complejidad algorítmica del problema.
4. Dada una cola de números enteros, **no** ordenada, construir un algoritmo que permita pasar a otra cola todos los elementos que no se repiten en la primera, sin destruir el contenido de la cola original y dejándola en su estado inicial. Determinar la complejidad algorítmica de la solución.

Ejemplo: si "C" contiene (12,6, 8, 5, 8, 12, 12) la cola resultado del proceso sería (6,5).

5. Dada una cola de valores enteros no repetidos y mayores o iguales a 2, obtener todos los valores que son Divisores Totales o parciales. Se dice que un valor es **Divisor Total** si permite dividir a **todos** los demás valores de la cola en forma exacta. Se dice que un divisor es **parcial** si al menos puede dividir en forma exacta al menos al **50%** de la cola

(es decir a la mitad de los elementos). Determinar la complejidad algorítmica de la solución.

Ejemplo: si “C” contiene (8, 12, 2, 6, 4) se dice que “2” es el divisor total de la cola porque divide al resto en forma exacta. Y “4” es divisor parcial por divide a 8,12 y el mismo.

6. Dada una pila y una cola generada con valores al azar retornar en una lista todos los valores comunes a ambas y en qué posición ordinal se encontró cada uno en su estructura. No se deben destruir las estructuras originales. No se deben perderse las estructuras originales. Determinar la complejidad algorítmica de la solución empleada.

Ejemplo: si “P” = (2,5,8,19,3,4) y “C” = (4, 18, 12, 5, 6) la lista tendría L = (5:2:4, 4:6:5).

7. Un negocio tiene 3 ventanillas para atender a sus clientes. Los clientes forman cola en cada ventanilla. Un día, dos de los tres empleados que atienden las ventanillas no pueden asistir al trabajo, quedando uno solo para atender a las tres colas. Este empleado decide que, a medida que lleguen los clientes, atenderá por cierta cantidad de minutos (que denominaremos Q) a cada cola, paseándose por todas las colas equitativamente.

Se pide que implemente un algoritmo que modele esta situación y dé como resultado el orden en que fueron atendidos los clientes.

Ejemplo: El algoritmo recibe un Q de tiempo que atenderá a cada cola y las tres colas cargadas con clientes, donde cada cliente es la cantidad de tiempo que se necesita para atenderlo.

Q = 10

Cola1 = (40, 20, 30)

Cola2 = (20, 10)

Cola3 = (10, 10, 10)

Resultado =

Cliente 1 Cola 3,

Cliente 1 Cola 2,

Cliente 2 Cola 3,

Cliente 2 Cola 2,

Cliente 3 Cola 3,

Cliente 1 Cola 1,

Cliente 2 Cola 1,

Cliente 3 Cola 1

IMPORTANTE !!!

- Los prácticos podrán ser resueltos en grupos de 4 **(tres) a 7 (seis)** alumnos. Condición indispensable para que los TPs sean evaluados.
- Todos los prácticos son obligatorios.
- Los prácticos deben ser entregados en tiempo y forma en las fechas que los docentes estipulen (serán determinadas según la cursada y avisadas mediante la plataforma virtual).



- Los prácticos NO se recuperan por lo que si son entregados en condiciones incorrectas de funcionamiento, fuera de término o fuera de formalidad (abajo descripto) se consideran desaprobados.
- Se deben presentar todos los ejercicios del tp. Se deben tener como mínimo un 60% de los ejercicios correctos para aprobar la práctica, con 2 de los 3 últimos ejercicios funcionando correctamente. Se evalúan tanto los errores de lógica como los de ingreso de datos. Deben controlar los ingresos de datos, tales como blancos, ingreso de números correctos, etc.
- Los prácticos deben presentarse con la formalidad:
 - Carátula Impresa o PDF por mail con: Fecha de Entrega, Integrantes de Grupo, Título del Práctico a Entregar y Nro de grupo si le fue asignado.
 - Se tomará asistencia en el momento de entrega del TP en caso presencial.
 - El código fuente debe estar disponible en GIT HUB, con acceso a los docentes.
 - Si se entregaron los prototipos (.h) de los ejercicios se deben respetar.
 - Los ejercicios deben ser posible probarlos en cualquiera de las implementaciones de la cola (arreglos o apuntadores).

El lenguaje de programación será “C”. (Usar el foro para consultas sobre el TP)

FECHA DE ENTREGA: 06/05/2024.

Profesor a Cargo de la Asignatura: Mario Perello.

Trabajos Prácticos a Cargo de: Claudia Reinaudi, José Racker, Pablo Chale y Mariano Goldman.