

4. Listas

Ejercicio 4.1

Diseñar un algoritmo para acceder, eliminar o insertar el k -ésimo elemento de una lista (siendo k una posición dada). Si la lista está vacía o si el valor de k esta fuera del rango del índice de la lista, invocar al procedimiento de ERROR. En cualquier otro caso, efectuar el procedimiento solicitado.

Ejercicio 4.2

Se dispone de una lista simplemente encadenada de números enteros, diseñar un algoritmo que a partir de ella genere otra lista conteniendo los nodos cuyos datos terminan en cero; dichos elementos, deberán ser eliminados de la lista original. Se asume que la lista está cargada, y que el algoritmo recibe como parámetro de entrada la dirección del primer elemento.

Ejercicio 4.3

Dada una lista simplemente encadenada de números diseñar un algoritmo que calcule en forma independiente: La suma de los números impares, y la suma de los números pares.

Ejercicio 4.4

Se dispone de una lista simplemente encadenada cuyos registros están ordenados en forma ascendente por una clave de tipo entero; diseñar un algoritmo que invierta el orden de la lista.

Ejercicio 4.5

Dada una lista simplemente encadenada que contiene datos de todas las provincias de la República Argentina: nombre, capital, cantidad total de habitantes y cantidad de analfabetos, y está ordenada en forma decreciente por número de habitantes analfabetos, generar otras tres listas que contengan el nombre, la capital y el porcentaje de analfabetos de las Provincias que respondan a las siguientes restricciones.

- L1: ≤ 10 % analfabetos
- L2: 16 a 25 % analfabetos
- L3: $\Rightarrow 26$ % analfabetos

Ejercicio 4.6

En el restaurante ÑOQUIS se está pensando en una solución informática para el soporte de datos del nuevo sistema de atención a clientes. Se han decidido por LISTAS por su dinamismo en cuanto a la cantidad de elementos. Diseñe un algoritmo que realice las siguientes funciones:

- Añadir cliente al ser atendido (lista simple ordenada por Nombre del Cliente).
- Registrar su consumo (Acumular el Total Consumido en valores de montos).
- Realizar el cobro (emitir ticket con Nombre, Fecha, Número de Mesa y Total).
- Eliminar del listado de atención.

La información almacenada debe mantenerse ordenada por Nombre del cliente.

Ejercicio 4.7

Genere un algoritmo que recorra una secuencia texto y genere una lista

simplemente encadenada con la frecuencia de utilización de cada letra. La lista debe mantenerse ordenada alfabéticamente y al final informar cual fue la frecuencia de cada letra y cuáles fueron la de mayor y menor frecuencia.

Ejercicio 4.8

Escribir un algoritmo que permita buscar, insertar o borrar un elemento identificado con una clave determinada en una lista circular simplemente encadenada.

Ejercicio 4.9

Dada una lista circular de 8 elementos que contienen, cada uno, un valor numérico entero para encriptación, encriptar un texto ingresado por teclado (arreglo de caracteres de 255 elementos como máximo). El texto encriptado debe almacenarse en una lista de salida simplemente encadenada y, por último mostrar por pantalla.

ENCRYPTAR

Multiplicador <small>N(5)</small>	Prox <small>Puntero</small>
-----------------------------------	-----------------------------

La encriptación se realiza según las siguientes pautas:

1. Dada la posición del carácter dentro del texto, buscar en la lista circular de encriptación el multiplicador correspondiente, de acuerdo a su posición lógica en la lista. Por ejemplo: como solo hay 8 multiplicadores, si se ingresa un arreglo de 10 caracteres, para encriptar los ultimos dos se utilizarían los multiplicadores 1 y 2.
2. Se usa la función ASCII(Carácter) para convertir cada carácter del arreglo en un entero.
3. Luego se lo multiplican los valores obtenidos en los 2 pasos anteriores.
4. Por último se descompone el resultado obtenido en 3 en cada uno de sus

dígitos, se los suma y se guarda en la lista de salida. Ej: si en el paso 3 me dio 123, $1+2+3=6$.

Ejercicio 4.10

Se dispone de una lista circular con un conjunto de números naturales. Este conjunto está dividido en 6 grupos. En cada grupo el primer elemento indica la cantidad de números subsiguientes a él que integran dicho grupo. Se desea generar una nueva lista doblemente encadenada en la cual cada nodo contendrá el promedio de cada grupo. Dicha lista debe quedar ordenada en forma ascendente y, al final se debe informar cuál fue el máximo y el mínimo promedio de todo el conjunto de números.

Ejercicio 4.11

En una empresa de servicios motorizados que NO posee una cantidad fija de integrantes, pero siempre es mayor que 1 (uno). Los datos de las personas está almacenados en un archivo (DNI, Nombre, Antigüedad) ordenado por DNI. Realizar un algoritmo que permita generar una lista de asignación (usando una estructura de lista doblemente encadenada), ordenada al principio por antigüedad, y un proceso que realice dicha asignación, en cada solicitud, por turno según orden en la lista. Una vez asignado dicha persona se va al final de la lista para esperar su nueva asignación. Al final del día informar cuantas asignaciones tuvo cada integrante.

PERSONAL

Nro_Personal <small>N(5)</small>	Nombre <small>AN(50)</small>	Asignaciones <small>N(5)</small>	Antigi
----------------------------------	------------------------------	----------------------------------	--------

Ejercicio 4.12

En el nuevo DELIVERY VAMOS-RAPIDO se está diseñando una solución

informática para la atención de clientes. Se ha pensado en una LISTA por su dinamismo en cuanto a la cantidad de elementos. Diseñe un algoritmo que realice las siguientes funciones:

- Registrar su pedido en una lista Doble (se agrega en Estado P (Pendiente), y se mantiene ordenado por Nombre del Cliente).
- Realizar el envío (cambia estado del pedido de P (pendiente) a E (Enviado)).
- Eliminar del listado de atención una vez recibido el cobro (al atender el cobro del cadete que hizo el trabajo).
- Datos a almacenar por pedido: Nombre, Dirección, Teléfono y Total.

La información almacenada debe mantenerse ordenada por Nombre del cliente.

Ejercicio 4.13

Dada una lista doblemente encadenada de enteros, ordenada en forma creciente, escriba un algoritmo que elimine las entradas repetidas. El algoritmo debe verificar al inicio que la lista que se pasa como argumento está ordenada; en caso afirmativo realizar la depuración y emitir un mensaje de 'ÉXITO'; en caso contrario, emitir un mensaje de 'ERROR'.

Ejercicio 4.14

Se desea ingresar una serie de números, e imprimirlos en el orden contrario al de ingreso. Diseñar un algoritmo que satisfaga tal requerimiento, utilizando la estructura de datos más apropiada.

Ejercicio 4.15

Supóngase que, utilizando una lista encadenada, se implementa una cola Q (estructura 'FIFO', es decir, 'primero en entrar, primero en salir'). Diseñar un algoritmo que permita insertar o extraer un elemento, comprobando siempre si

la cola está vacía o no.

Ejercicio 4.16

Supóngase que, utilizando una lista encadenada, se implementa una pila P (estructura 'LIFO', es decir, 'último en entrar, primero en salir'). Diseñar un algoritmo que permita insertar o extraer un elemento, comprobando siempre si la pila está vacía o no.

Ejercicio 4.17

Teniendo en cuenta el ejercicio 1.20 del TP2:

Se posee 2 secuencias (S1 y S2) con las cuales se desea generar una nueva secuencia (SAL) donde se intercalen las palabras de las secuencias de entrada, de la siguiente manera: copiar de S1 aquellas palabras que empiezan y terminan con la misma letra y de S2 aquellas palabras que posean al menos un dígito numérico y además estén en posición par.

Escribir un algoritmo que lo resuelva, teniendo en cuenta que conoce la estructura de listas.