

**Instituto Tecnológico de Tijuana**

**Subdirección Académica**

**Departamento de Sistemas y Computación**

**Semestre agosto-diciembre 2018**

**Carrera:** Ing. En Sistemas Computacionales Serie SC1A

**Materia:** Estructura de Datos

**Unidad 3 – Listas Circulares**

**Alumno:**15211883 - Angeles Valadez Jonathan

**Profesor:** Ray Brunett Parra Galaviz

**Listas circulares**

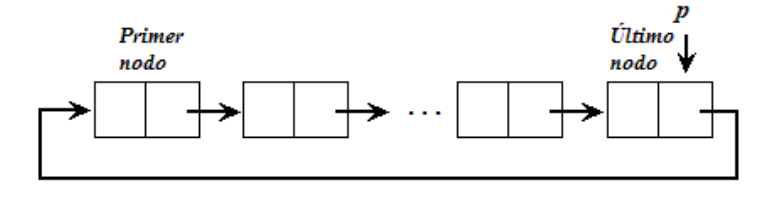
Algunos de los problemas que se tienen con una lista simplemente ligada son:

* Dado un apuntador ptr para un nodo en una lista lineal, no se puede llegar a ninguno de los nodos que preceden a ptr.
* Si se recorre la lista, el apuntador externo a la lista debe preservarse para que se pueda hacer referencia a la lista otra vez.

Suponga que se hace un pequeño cambio a la estructura de una lista lineal:

* El campo next en el último nodo contenga un apuntador de regreso al primer nodo y no un apuntador nulo.

Al tipo de lista que adopta este cambio se le denomina lista circular.



En una lista circular es posible llegar a cualquier otro punto de ella partiendo de un nodo determinado.

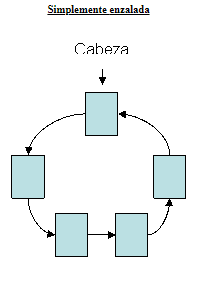
Debe observarse que una lista circular no tiene un “primer” o “último” nodo natural, por lo que se debe establecer un primer y último nodo por convención. Otra convención útil, es permitir que el apuntador externo a la lista circular apunte al último nodo y permitir que el siguiente nodo sea el primero de la lista, tal y como lo muestra la Ilustración. También se establece la convención de que un apuntador nulo representa una lista circular vacía.

**¿Por qué utilizar una estructura Circular?**

Imaginemos por un segundo una lista simplemente enlazada, el movimiento siempre fluirá desde la cabeza en dirección hacia el final de la lista, pero ¿qué ocurre cuando desde el último nodo se necesita operar con el primero?, este es el punto diferencial de una estructura abierta (aquella en que algún nodo apunta a null) y una cerrada.

**En una lista circular:**

* No existe algún elemento que apunte a NULL
* Se integra una estructura tipo anillo
* Solo hay una cabeza
* La cabeza siempre será el siguiente enlace para algún nodo
* Se pueden llegar a crear recorridos en bucles infinitos



**Operaciones:**

***CreaNodo***

* ***Descripción:*** Realiza la solicitud del espacio de almacenamiento para almacenar un nodo***.***
* ***Datos de ingreso:*** Apuntador a Nodo.
* ***Datos de salida:*** Espacio de almacenamiento referenciado por el apuntador a Nodo.
* ***Precondición:*** Apuntador a Nodo se encuentra inicializado a NULL.
* ***Postcondición:*** Apuntador a Nodo poseerá una dirección de memoria, si es nula se interpretará como la imposibilidad de obtener espacio de almacenamiento, por lo que el objeto no se creará.

***InsertaInicio***

* ***Descripción:*** Inserta un nodo en la posición referenciada por el apuntador ***Cabeza.***
* ***Datos de ingreso:*** Apuntador a Nodo.
* ***Datos de salida:*** Apuntador a Cabeza.
* ***Precondición:*** True.
* ***Postcondición:*** Lista poseerá uno o más elementos.  Cabeza apuntará a una dirección física.

***InsertaAntes***

* ***Descripción:*** Inserta un nodo en la posición previa a la que posee el carácter de ingreso.  Actualiza la estructura Lista.
* ***Datos de ingreso:*** Apuntador Cabeza, Apuntador a Nodo, Caracter.
* ***Datos de salida:*** Apuntador a Cabeza, o mensaje de imposibilidad de ingresar el nodo, dada la ausencia en la estructura de aquel nodo que posea como elemento al Carácter buscado.
* ***Precondición:*** True
* ***Postcondición:*** Lista podrá poseer un elemento más.

***InsertaDespues***

* ***Descripción:*** Inserta un nodo en la posición posterior a la que posee el carácter de ingreso.  Actualiza a estructura Lista.
* ***Datos de ingreso:*** Apuntador Cabeza, Apuntador a Nodo, Caracter.
* ***Datos de salida:*** Apuntador a Cabeza, o mensaje de imposibilidad de ingresar el nodo, dada la ausencia en la estructura de aquel nodo que posea como elemento al Carácter buscado.
* ***Precondición:*** True
* ***Postcondición:*** Lista podrá poseer un elemento más.

***VisualizaLista***

* ***Descripción:*** Permite verificar los elementos de la Lista, partiendo de la Cabeza
* ***Datos de ingreso:*** Apuntador a Cabeza.
* ***Datos de salida:*** Ninguno.
* ***Precondición:*** El usuario ha seleccionado la cabeza desde la cual hará la visualización.
* ***Postcondición:*** De ninguna manera la estructura será actualizada.

***EliminaNodo***

* ***Descripción:*** Permite prescindir del nodo que posee el Carácter de ingreso (si existe), actualizando la estructura.
* ***Datos de ingreso:*** Apuntador Cabeza, Carácter.
* ***Datos de salida:*** Cabeza con estructura modificada, entero 1: fue eliminado, 0: no eliminado.
* ***Precondición:*** True
* ***Postcondición:*** De encontrarse el nodo con el carácter buscado, Lista poseerá un elemento menos, de manera contraria se retornará valor para emitir mensaje de nodo no encontrado.

***BuscaNodo***

* ***Descripción:*** Recorre la estructura hasta alcanzar el nodo que contiene el carácter buscado.
* ***Datos de ingreso:*** Apuntador Cabeza, Carácter.
* ***Datos de salida:*** 1 si el nodo ha sido encontrado, 0 de manera contraria.
* ***Precondición:*** True
* ***Postcondición:*** De ninguna manera la estructura será actualizada.

***OrdenaLista***

* ***Descripción:*** Actualiza la lista, teniendo en cuenta un orden ascendente según los nodos que se tienen.
* ***Datos de ingreso:*** Apuntador Cabeza.
* ***Datos de salida:*** Apuntador a Cabeza y estructura modificada.
* ***Precondición:*** True
* ***Postcondición:*** Se contará al término de la operación con todos los nodos que se tenían a su inicio.

***CuentaNodos***

* ***Descripción:*** determina la longitud de la estructura a partir del número de nodos que posee Lista.
* ***Datos de ingreso:*** Apuntador Cabeza.
* ***Datos de salida:*** Número entero igual o superior a cero.
* ***Precondición:*** True
* ***Postcondición:*** La estructura no será modificada.

**Bibliografía**

[En Línea]: <http://www.utm.mx/~rruiz/cursos/ED/material/listas.pdf>

[En Línea]: <http://artemisa.unicauca.edu.co/~mendoza/Laboratorio6_LabEdatos1.htm>