**3.3 - Listas**

**Qué es una Lista**

La Lista es una estructura de datos muy importante en los lenguajes de programación donde, representa una colección de elementos ordenados, puede contener elementos repetidos y cada elemento de la lista tiene un índice que lo ubica dentro de la misma.

**Clasificaciones de Listas (simplemente enlazada, doblemente enlazada, circulares).**

**Listas simplemente enlazadas:** Una lista simplemente enlazada pertenece a las estructuras de datos fundamentales. Suele utilizarse para implementar otras estructuras de datos. Está estructurada en una secuencia de nodos, en los que se guardan los datos  y un puntero que apunta (contiene la dirección de la ubicación) al siguiente nodo.

**Listas doblemente enlazadas:** [^](http://c.conclase.net/edd/?cap=005#inicio)Una lista doblemente enlazada es una lista lineal en la que cada nodo tiene dos enlaces, uno al nodo siguiente, y otro al anterior. Las listas doblemente enlazadas no necesitan un nodo especial para acceder a ellas, pueden recorrerse en ambos sentidos a partir de cualquier nodo, esto es porque a partir de cualquier nodo, siempre es posible alcanzar cualquier nodo de la lista, hasta que se llega a uno de los extremos.

**Listas circulares:** Una lista circular es una lista lineal en la que el último nodo a punta al primero. Las listas circulares evitan excepciones en las operaciones que se realicen sobre ellas.

**Cuáles son las operaciones básicas de una lista.**

* Añadir un elemento a una lista doblemente enlazada vacía.
* Insertar un elemento en la primera posición de la lista.
* Insertar un elemento en la última posición en la lista.
* Insertar un elemento a continuación de un nodo cualquiera de una lista.
* Buscar o localizar elementos.
* Borrado de elementos.
* Eliminar único elemento de una lista doblemente enlazada
* Eliminar el primer elemento de la lista doblemente enlazada.
* Eliminar el último elemento de una liara doblemente enlazada.
* Eliminar un elemento intermedio de una lista doblemente enlazada.

**Ventajas:**

* Las listas son dinámicas, es decir, se pueden almacenar en ellas tantos elementos como se necesiten, siempre y cuando haya espacio suficiente en la memoria de la computadora.
* Al insertar un elemento en la lista, la operación tiene un tiempo constante independientemente de la posición en la que se inserte, solo se debe crear el nodo y modificar los enlaces de los mismos.
* Al eliminar un elemento sucede lo mismo que se mencionó en el punto anterior.

**Desventajas:**

* El acceso a un elemento es más lento, debido a que la información no está en posiciones contiguas en la memoria de la computadora, por lo que no se puede acceder a un elemento con base en su posición como se hace en los arreglos

**Programa ejemplo en C# de un TDA Lista.**

**3.3 – Listas**

<https://sites.google.com/site/programacioniiuno/temario/unidad-3---estructuras-de-datos-comunes-y-colecciones/la-estructura-lista>

<https://analisisyprogramacionoop.blogspot.com/2017/07/lista-simplemente-enlazada-C-sharp.html>

<http://c.conclase.net/edd/?cap=005>

<http://estr-datos-omar.blogspot.com/2011/10/listas-circulares.html#:~:text=Una%20lista%20circular%20es%20una,uno%20anterior%20y%20uno%20siguiente>

<http://estructura-de-datos-u3.blogspot.com/2014/10/311-operaciones-basicas-con-listas.html>

<https://easynetstudio.wixsite.com/easynetstudio/listas>

[Documentacion\_Biblioteca\_Estructuras\_Datos\_Avanzadas.pdf (informatica-juridica.com)](https://www.informatica-juridica.com/wp-content/uploads/2014/01/Documentacion_Biblioteca_Estructuras_Datos_Avanzadas.pdf)