

TAREA #12 LISTAS ENLAZADAS SIMPLES

CINTHIA GUADALUPE OLIVAS CALDERON NO.17212165

Una lista es una estructura de datos dinámica que contiene una colección de elementos homogéneos, con una relación lineal entre ellos. Una relación lineal significa que cada elemento (a excepción del primero) tiene un precedente y cada elemento (a excepción del último) tiene un sucesor. Gráficamente, una lista se puede conceptualizar de la siguiente forma:



Una lista lineal es un conjunto de elementos de un tipo dado que pueden variar en número y donde cada elemento tiene un único predecesor y un único sucesor o siguiente, excepto el primero y último de la lista. Esta es una definición muy general que incluye los ficheros y vectores.

Los elementos de una lista lineal se almacenan normalmente contiguos —un elemento detrás de otro— en posiciones consecutivas de la memoria. Las sucesivas entradas en una guía o directorio telefónico, por ejemplo, están en líneas sucesivas, excepto en las partes superior e inferior de cada columna. Una lista lineal se almacena en la memoria principal de una computadora en posiciones sucesivas de memoria; cuando se almacenan en cinta magnética, los elementos sucesivos se presentan en sucesión en la cinta. Esta asignación de memoria se denomina almacenamiento secuencial.

Las líneas así definidas se denominan contiguas.

Las operaciones que se pueden realizar con listas lineales contiguas son:

1. Insertar, eliminar o localizar un elemento.
2. Determinar el tamaño —número de elementos— de la lista.
3. Recorrer la lista para localizar un determinado elemento.

4. Clasificar los elementos de la lista en orden ascendente o descendente.
5. Unir dos o más listas en una sola.
6. Dividir una lista en varias sublistas.
7. Copiar la lista.
8. Borrar la lista.

Una lista lineal contigua se almacena en la memoria de la computadora en posiciones sucesivas o adyacentes y se procesa como un array unidimensional. En este caso, el acceso a cualquier elemento de la lista y la adición de nuevos elementos es fácil; sin embargo, la inserción o borrado requiere un desplazamiento de lugar de los elementos que le siguen y, en consecuencia, el diseño de un algoritmo específico.

Para permitir operaciones con listas como arrays se deben dimensionar éstos con tamaño suficiente para que contengan todos los posibles elementos de la lista.

Estas listas están formadas por un conjunto de nodos, en los que cada elemento contiene un puntero con la posición o dirección del siguiente elemento de la lista, es decir su enlace. Se dice entonces, que los elementos de una lista están enlazados por medio de los campos enlaces. Cada nodo está constituido por dos partes: **información o campos de datos** (uno o varios) y un **puntero** (con la dirección del nodo siguiente). Al campo o campos de datos del nodo lo designaremos como INFORMACIÓN del nodo y al puntero por SIGUIENTE del nodo. Con esta organización de los datos, es evidente que no es necesario que los elementos de la lista estén almacenados en posiciones físicas adyacentes para estar relacionados entre sí, ya que el puntero indica unívocamente la posición del dato de la siguiente lista.

Para definir una lista enlazada, además de la estructura de cada uno de sus nodos, necesitamos una variable externa a la propia lista, con un puntero que marque la posición de la cabeza (inicio, primero) de la lista. Esta variable es quien normalmente da nombre a la lista, pues nos dice donde localizarla, sea cual sea su tamaño. Para detectar el último elemento de la lista se emplea un puntero nulo, que por convenio, se suele representar de diversas formas: por un enlace con la palabra reservada nil (NULO), por una barra

inclinada (/) o por un signo especial, tomado de la toma de tierra en electricidad. Una lista enlazada sin ningún elemento se llama lista vacía y las distinguiremos asignando a su puntero de cabecera el valor nil

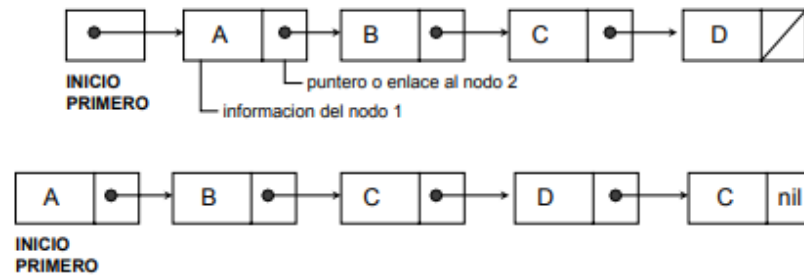


Fig. 5.5. *Ejemplos de listas enlazadas*

REFERENCIAS

- Nell Dale, Object-oriented Data Structures using Java, Jones and Bartlett Publishers, 2002.
- Bottazzi, C., & Costarelli, S., & Dalcin, L.. (2017). Tipos de datos abstractos fundamentales. En Algoritmos y Estructuras de Datos(304). Universidad Nacional del Litoral: Centro de Investigacion de M ´etodos Computacionales.
- Joyanes Aguilar, L., & Zahonero Martínez, I. (1998). Estructura de datos. Algoritmos, abstracción y objetos. (Ed. rev.). Madrid, España: McGraw-Hill