

Estructuras de Datos Elementales

Pilas

La principal característica de este tipo de estructura es que la última variable añadida en la estructura es también la primera que saldrá. Suele relacionarse con una pila de platos vertical, donde para poder obtener un plato en concreto, es necesario remover todos los que estén encima de él para poder sacarlo. Generalmente la palabra empleada para ordenar los datos de esta manera es la de apilar o empujar.

Filas

La operación para insertar datos en las filas se llama formar en la fila, o *enqueue* y para retirarlos se es *dequeue*. Las filas poseen una cabeza y una cola, a diferencia de las pilas que solo poseen una cabeza. El modo de insertar y retirar datos nos dice que el último dato en ingresarse es el primer dato en salir.

Listas Ligadas

Una lista ligada es una estructura de datos en la que los objetos están ordenados de manera lineal. A diferencia de los arreglos, cuyos datos están ordenados según el índice de estos arreglos, las listas se encuentran ordenados por medio de apuntadores.

Estas listas pueden tener distintas formas. Una de ellas es la lista ligada simple, cuyos datos se encuentran unidos mediante un puntero uno tras otro. Los datos de las listas ligadas dobles se encuentran unidos entre ellos, apuntando tanto al anterior como al siguiente.

Las ligas en anillo se encuentran ligadas entre todas, de esta forma, el dato final se encuentra ligado con el dato del inicio, dándole esta forma característica de anillo.

A continuación, se muestran algunas de las operaciones básicas con las listas ligadas:

- Buscar dentro de una lista ligada. Esta operación se realiza mediante ciclos. Se define una variable que contenga el dato que se va a buscar y el ciclo debe recorrer todas las partes de la lista hasta que el apuntador apunte a un espacio nulo. Todos los datos contenidos en las celdas se compararán con el dato contenido en la variable para ver si existen. De ser así, se hace saber directamente al usuario.
- Insertar dentro de una lista ligada. Para realizar esta operación, debe crearse una función de tipo *void* en la que, por cada nuevo dato insertado, se creará una nueva celda que se unirá a la lista.
- Borrar dentro de una lista ligada. En este caso, lo que debe hacerse es crear una variable que contenga el dato que desea borrarse, mediante un ciclo se recorrerán todas las celdas de la lista hasta encontrar un dato que se parezca al dato contenido en la variable. Después, la celda anterior se liga con la celda siguiente de la celda a eliminar y viceversa de poseerse una lista ligada doble, y de ser el fin de la lista, el apuntador a siguiente de la celda anterior apuntará a un espacio nulo.

Una representación de objetos de array múltiple

Es posible representar distintos tipos de datos en una misma estructura de datos. Para ello, es necesario crear una estructura que contenga un array diferente por cada tipo de dato presente en la estructura que planea almacenarse.

Una representación de objetos de array simple

Todos los objetos que se pueden procesar en una computadora poseen un tamaño de entre 0 a $M-1$ donde M es un número entero grande de tipo *long int*. Cada uno de los bits de ese dato se almacenan en forma de un arreglo. En gran parte de los lenguajes de programación, un puntero apunta hacia donde se encuentra el primer bit del dato, y cada una de las celdas de memoria apuntan hacia donde se encuentra el siguiente.

Almacenando y liberando objetos

Para insertar un dato dentro de una lista ligada, es necesario crear una nueva celda que se encuentre ocupada por algún objeto en desuso por la computadora.

Es una forma efectiva de administrar los objetos en desuso del almacenamiento. Para liberar celdas de memoria que ya no se utilizarán, se guarda en una nueva lista ligada.

Representación de árboles enraizados de ramificación ilimitada

Se puede expandir el esquema para representar un árbol binario donde el número de *child* es una constante ***k***. Generalmente este tipo de árboles se emplea cuando el número de hijos de un nodo es desconocido, pues no conocemos la cantidad de atributos que poseen.