



Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

Laboratorios de computación salas A y B

M.I. Marco Antonio Martínez Quintana

Profesor:

Asignatura:

Estructura de datos y algoritmos I

Grupo:

15

No de Práctica(s):

Práctica 6: Estructura de Datos – Cola y Listas

Integrante(s):

Carrasco Ruiz Alan Uriel

*No. de Equipo de
cómputo empleado:*

/

No. de Lista o

/

Semestre:

2021-2

Fecha de entrega:

09 de Agosto del 2021

Observaciones:

CALIFICACIÓN: _____

Práctica 5 – Colas:

Objetivo:

Revisarás las definiciones, características, procedimientos y ejemplos de las estructuras lineales Pila y Cola, con la finalidad de que comprendas sus estructuras y puedas implementarlas.

Introducción:

Los conjuntos (colecciones de datos) son tan fundamentales para las ciencias de la computación como lo son para las matemáticas.

Una estructura de datos consiste en una colección de nodos o registros del mismo tipo que mantienen relaciones entre sí. Un nodo es la unidad mínima de almacenamiento de información en una estructura de datos.

Las estructuras de datos lineales son aquellas en las que los elementos ocupan lugares sucesivos en la estructura y cada uno de ellos tiene un único sucesor y predecesor.

Cola

La cola (o queue) es una estructura de datos lineal, en la cual el elemento obtenido a través de la operación ELIMINAR está predefinido y es el que se encuentra al inicio de la estructura.

La cola implementa la política First-In, First-Out (FIFO), esto es, el primer elemento que se agregó es el primero que se elimina.

La cola es una estructura de datos de tamaño fijo y cuyas operaciones se realizan por ambos extremos; permite INSERTAR elementos al final de la estructura permite

ELIMINAR elementos por el inicio de esta. La operación de INSERTAR también se le llama ENCOLAR y la operación de ELIMINAR también se le llama DESENCOLAR.

Para poder diseñar un algoritmo que defina el comportamiento de una COLA se deben considerar 3 casos para ambas operaciones (INSERTAR y ELIMINAR):

- ♣ Estructura vacía (caso extremo).
- ♣ Estructura llena (caso extremo).
- ♣ Estructura con elemento(s) (caso base).

Cola vacía:

La cola posee dos referencias, una al inicio (HEAD) y otra al final (TAIL) de la cola. En una cola vacía ambas referencias (HEAD y TAIL) apuntan a nulo. En una cola vacía no es posible desencolar debido a que la estructura no posee elementos. En una cola vacía sí se pueden encolar elementos, en este caso las referencias HEAD y TAIL apuntan al mismo elemento, que es el único en la estructura.

Cola llena:

Cuando la referencia a tail de una cola llega a su máxima capacidad de almacenamiento (MAX) se dice que la cola está llena. En una cola llena no es posible encolar más elementos. En una cola llena sí se pueden desencolar elementos, en tal caso se obtiene el elemento al que hace referencia head y esta referencia se recorre al siguiente elemento (sucesor).

Cola con datos:

Una cola que contiene elementos (sin llegar a su máximo tamaño) representa el caso general de la estructura. En una cola con elementos es posible desencolar nodos, recorriendo la referencia al inicio de la cola (HEAD) al siguiente elemento de la estructura. Así mismo, se pueden encolar elementos en una cola mientras la referencia al final (TAIL) de la estructura no sea mayor al tamaño máximo de la misma. Cuando se encola un elemento, el nodo al que apunta TAIL tiene como sucesor el nuevo elemento y la referencia a TAIL apunta al nuevo elemento.

Aplicaciones de las colas:

1. La cola de impresión permite enviar documentos de gran tamaño, o varios documentos, a una impresora sin tener que esperar que se complete la impresión para seguir con la siguiente tarea. Cuando se envía un archivo a imprimir, se crea un archivo de almacenamiento intermedio en formato EMF, donde se almacena lo que se envía a la impresora y las opciones de

impresión. Las impresiones se van realizando según vayan llegando los archivos (FIFO).

2. Las colas se utilizan para asignar tiempo a los distintos usuarios de los dispositivos de entrada/salida (E/S), impresoras, discos, cintas y demás. El sistema operativo mantiene colas para peticiones de imprimir, leer o escribir en cada uno de estos dispositivos.
3. Las colas se utilizan en sistemas informáticos, transportes y operaciones de investigación (entre otros), donde los objetos, personas o eventos son tomados como datos que se almacenan y se guardan mediante colas para su posterior procesamiento.

Práctica 6 – Listas:

Objetivo:

Revisarás las definiciones, características, procedimientos y ejemplos de las estructuras lineales Lista simple y Lista circular, con la finalidad de que comprendas sus estructuras y puedas implementarlas.

Introducción:

Las listas son un tipo de estructura de datos lineal y dinámica. Es lineal porque cada elemento tiene un único predecesor y sucesor, y es dinámica porque su tamaño no es fijo y se puede definir conforme se requiera. Las operaciones básicas dentro de una lista son BUSCAR, INSERTAR Y ELIMINAR.

Lista simple:

Una lista simple (también conocida como lista ligada o lista simplemente ligada) está constituida por un conjunto de nodos alineados de manera lineal (uno después de otro) y unidos entre sí por una referencia.

A diferencia de un arreglo, el cual también es un conjunto de nodos alineados de manera lineal, el orden está determinado por una referencia, no por un índice, y el tamaño no es fijo.

La unidad básica de una lista simple es un elemento o nodo. Cada elemento de la lista es un objeto que contiene la información que se desea almacenar, así como una referencia (NEXT) al siguiente elemento (SUCEsor).

Para poder diseñar un algoritmo que defina el comportamiento de una LISTA LIGADA se deben considerar 2 casos para cada operación (BUSCAR, INSERTAR y ELIMINAR):

- ♣ Estructura vacía (caso extremo).
- ♣ Estructura con elemento(s) (caso base).

Buscar:

El método debe buscar el primer elemento que coincida con la llave K dentro de la lista L, a través de una búsqueda lineal simple, regresando un apuntador a dicho elemento si éste se encuentra en la lista o nulo en caso contrario.

Una lista simple vacía no contiene elementos, la referencia al inicio de la misma (head) apunta a nulo, por lo tanto, en una lista vacía no es posible buscar elementos.

Una lista simple con elementos puede contener de 1 a n elementos, en tal caso, la referencia al inicio (HEAD) apunta al primer elemento de la lista. Es posible recorrer la lista a través de la referencia (NEXT) de cada nodo hasta llegar al que apunta a nulo, el cuál será el último elemento. Por lo tanto, dentro de una lista simple con elementos es posible buscar una llave K.

Insertar:

Dado un nodo x que contenga una llave k previamente establecida, el método INSERTAR agrega el elemento x al inicio de la lista.

Es posible insertar elementos tanto en una lista simple vacía como en una lista simple con elementos. Cuando se inserta un nuevo elemento en una lista simple vacía la referencia al inicio de la lista (HEAD) apunta al nodo insertado.

Cuando se inserta un nuevo elemento en una lista simple con elementos, la referencia del nuevo nodo (NEXT) apunta al mismo nodo al que apunta el inicio de la lista (HEAD) y ahora HEAD apunta al nuevo nodo.

Borrar:

El método elimina el elemento x de la lista L (si es que éste se encuentra en la estructura).

Para eliminar un elemento de la lista primero es necesario saber la ubicación del nodo a eliminar, por lo tanto, primero se debe realizar una búsqueda del elemento.

En una lista simple vacía no es posible eliminar, debido a que esta estructura no contiene elementos.

Para eliminar un nodo en una lista simple con elementos, primero se debe buscar el elemento a eliminar, una vez encontrado el nodo en la lista, se deben mover las referencias de la estructura de tal manera de que el antecesor del nodo a eliminar apunte al sucesor de este.

Aplicaciones:

1. Cualquier red social utiliza una lista simple, en la que cada elemento tiene un único sucesor que sería la siguiente publicación, hasta llegar a la última.
2. Un registro de alumnos de algún salón podría ser ingresados manualmente por apellidos, y esta los iría almacenando uno después de otro, para que no se mezclaran.
3. Otra aplicación es para las bases de datos que son muy usadas en la programación, ya que se puede mantener la información en forma de listas, y si queremos trabajar datos específicos de las listas, lo almacenamos en nuevas listas para su manipulación.

Conclusiones:

Las colas y listas son fundamentales, para el manejo de datos, como para el almacenamiento de una forma ordenada, en la programación son muy importantes, ya que en proyectos más grandes se requieren de bases de datos que podemos hacer fácilmente con las listas, las colas nos permitirán sacar datos de forma que el ultimo dato que metamos será el primero en salir, el conocer estos temas es importante y beneficioso para los ingenieros en computación ya que manejar esta parte es un plus para otros ingenieros que solo llevan las bases.

Bibliografía:

es.wikipedia.org/wiki/Wikipedia:Uso_de_listas

<http://lcp02.fi-b.unam.mx/>

