



TECNOLÓGICO NACIONAL DE MÉXICO

CAMPUS JEREZ

MATERIA: ESTRUCTURA DE DATOS

DOCENTE: ISC SALVADOR ACEVEDO SANDOVAL

3° SEMESTRE

TEMA 3: ESTRUCTURAS LINEALES

ACTIVIDAD 1: MAPA CONCEPTUAL

ALUMNA: LIZA AREMY SANTANA CONTRERAS

NO. CONTROL: 16070005

18 DE DICIEMBRE DE 2020

JEREZ DE GARCIA SALINAS

RESUMEN

PILAS

¿Qué son?

Es una estructura de datos que almacena y recupera sus elementos atendiendo a un orden estricto. Las pilas se conocen también como estructuras LIFO (Last-in, first-out, último en entrar primero en salir).

Una pila (Stack) es una estructura de datos de entradas ordenadas que sólo se pueden introducir y eliminar por un extremo, llamado cima.

Características de operación

- Evaluación de expresiones en notación postfija (notación polaca inversa).
- Reconocedores sintácticos de lenguajes independientes del contexto
- Implementación de recursividad.

Representación en memoria

Las pilas no son estructuras de datos fundamentales, es decir, no están definidas como tales en los lenguajes de programación. Las pilas pueden representarse mediante el uso de:

- Arreglos.
- Listas enlazadas.

Operaciones básicas

Las operaciones usuales en la pila son Insertar y Quitar. La operación Insertar (push) añade un elemento en la cima de la pila, y la operación Quitar (pop) elimina o saca un elemento de la pila.

La operación Insertar (push) sitúa un elemento dato en la cima de la pila, y Quitar (pop) elimina o quita el elemento de la pila.

Tipo de dato	Dato que se almacena en la pila
Operaciones	
Crear pila	Inicia
Insertar (push)	Pone un dato en la pila
Quitar(pop)	Retira (saca) un dato de la pila
Pila vacía	Comprueba si la pila no tiene elementos
Pila llena	Comprueba si la pila está llena de elementos
Limpiar pila	Quita todos sus elementos y deja la pila vacía.
<i>Cima pila</i>	Obtiene el elemento cima de la pila
<i>Tamaño de la pila</i>	Número de elementos máximo que puede contener la pila.

Aplicaciones reales

Las pilas se utilizan en compiladores, sistemas operativos y programas de aplicaciones. Una aplicación interesante es la evaluación de expresiones aritméticas, también la organización de la memoria.

NOTA:

Los elementos que forman la pila se guardan en arrays (arreglos), en el contenedor Vector o bien formando una lista enlazada. La implementación con un array (arreglo) es estática ya que el array es de tamaño fijo. La clase PilaLineal, con esta representación, necesita un array y una variable numérica, cima, que apunte al último elemento colocado en la pila. Al utilizar un array es necesario tener en cuenta

que el tamaño de la pila no puede exceder el número de elementos del array, y la condición pila llena será significativa para el diseño.

Para utilizar una pila de elementos de tipo primitivo (int, char, long, float, double...) es necesario, para insertar, crear un objeto de la correspondiente clase envolvente (Integer, Character, Long, Float, Double...) y pasar dicho objeto como argumento del método insertar().

COLAS

¿Qué son?

Esta estructura, al igual que las pilas, almacena y recupera sus elementos atendiendo a un orden estricto. Las colas se conocen como estructuras FIFO (first-in, first-out, primero en entrar-primero en salir), debido a la forma y orden de inserción y de extracción de elementos.

Una cola es una estructura de datos que almacena elementos en una lista y permite acceder a los datos por uno de los dos extremos de la lista. Un elemento se inserta en la cola (parte final) de la lista y se suprime o elimina por el frente (parte inicial, frente) de la lista.

Representación en memoria

Podemos representar a las colas de dos formas:

- Como arreglos.
- Como listas ordenadas.

En esta unidad trataremos a las colas como arreglos de elementos, en donde debemos definir el tamaño de la cola y dos apuntadores, uno para acceso al primer elemento de la lista y otro que guarde el último. En lo sucesivo, al apuntador del primer elemento lo llamaremos F, al del último elemento A y MAXIMO para definir el número máximo de elementos en la cola.

Operaciones básicas

Tipo de dato	Elemento que se almacena en la cola
Operaciones	
Crear cola	Inicia la cola como vacía
Insertar	Añade un elemento por el final de la cola
Quitar	Retira (extrae) el elemento de la cola
Cola vacía	Comprueba si la cola no tiene elementos
Cola llena	Comprueba si la cola está llena de elementos
Frente	Obtiene el elemento frente o primero de la cola
Tamaño de la cola	Número de elementos máximo que puede contener la cola.

La operación insertar toma un elemento y lo añade al final de la cola. quitar suprime y devuelve el elemento de la cabeza de la cola. La operación frente devuelve el elemento que está en la primera posición (frente) de la cola, sin eliminar el elemento. La operación de control colaVacía comprueba si la cola tiene elementos, ya que es necesaria esta comprobación antes de eliminar un elemento. La operación colaLlena comprueba si la cola está llena, esta comprobación se realiza antes de insertar un nuevo miembro. Si las precondiciones para insertar y quitar se violan, el programa debe generar una excepción o error.

Aplicaciones reales

Las colas tienen numerosas aplicaciones en el mundo de la computación: colas de mensajes, colas de tareas a realizar por una impresora, colas de prioridades, etc.

Las aplicaciones utilizan una cola para almacenar elementos en su orden de aparición o concurrencia.

Numerosos modelos de sistemas del mundo real son de tipo cola: la cola de impresión en un servidor de impresoras, los programas de simulación, las colas de prioridades en organización de viajes. Una cola es la estructura típica que se suele utilizar como almacenamiento de datos, cuando se envían datos desde un componente rápido de una computadora a un componente lento (por ejemplo, a una impresora).

Tipos

Colas de prioridad

Esta estructura es un conjunto de elementos donde a cada uno de ellos se les asigna una prioridad, y la forma en que son procesados es la siguiente:

- Un elemento de mayor prioridad es procesado al principio.
- Dos elementos con la misma prioridad son procesados de acuerdo al orden en que fueron insertados en la cola.

Cola con array circular.

Las colas lineales tienen un grave problema, como las extracciones sólo pueden realizarse por un extremo, puede llegar un momento en que el apuntador A sea igual al máximo número de elementos en la cola, siendo que al frente de la misma existan lugares vacíos, y al insertar un nuevo elemento nos mandará un error de overflow (cola llena). Para solucionar el problema de desperdicio de memoria se implementaron las colas circulares, en las cuales existe un apuntador desde el último elemento al primero de la cola.

Las condiciones que debemos tener presentes al trabajar con este tipo de estructura son las siguientes:

- Over flow, cuando se realice una inserción.
- Under flow, cuando se requiera de una extracción en la cola.
- Vacio.

Bicolas: Colas de doble entrada

Esta estructura es una cola bidimensional en que las inserciones y eliminaciones se pueden realizar en cualquiera de los dos extremos de la bicola. Gráficamente representamos una bicola de la siguiente manera:

Existen dos variantes de la doble cola:

- Doble cola de entrada restringida.
- Doble cola de salida restringida.

La primera variante sólo acepta inserciones al final de la cola, y la segunda acepta eliminaciones sólo al frente de la cola.

Bicolas con listas enlazadas.

LISTAS

¿Qué son?

Una lista es una colección de elementos llamados generalmente nodos. El orden entre los nodos se establece por medio de apuntadores (realizando el enlace entre nodos), es decir, direcciones o referencias a otros nodos.

Representación en memoria

Los tipos de datos abstractos más conocidos son la pila, la lista ordenada, la lista doblemente enlazada ordenada y el árbol binario. Y son un ejemplo claro del manejo de memoria en forma dinámica.

Operaciones básicas

Las operaciones más comunes son:

- Recorrido de la lista.
- Inserción de un elemento.
- Borrado de un elemento.
- Búsqueda de un elemento.

Aplicaciones reales

Dos de las aplicaciones más conocidas de listas son:

- Representación de polinomios
- Resolución de colisiones (Hash)

En general puede decirse que las listas son muy útiles para aquellas aplicaciones en las cuales se necesite dinamismo en el crecimiento y reducción de las estructuras de datos.

Aplicación´ de Listas: Recolección´ de Basura y Compactaciones un ambiente de computadores multiprocesos, muchos programas residen en memoria al mismo.

Tipos

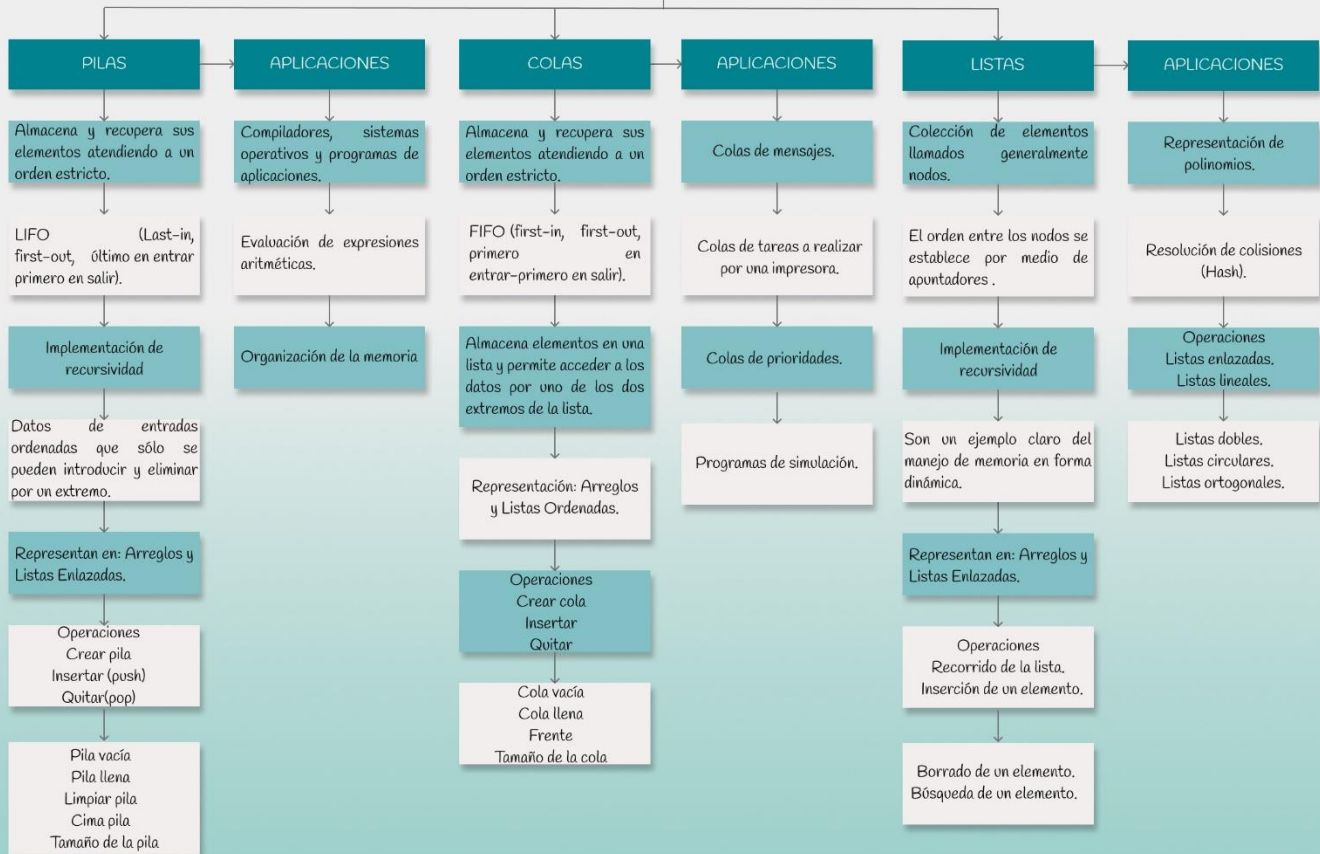
- Listas enlazadas.
- Listas lineales.
- Listas dobles.
- Listas circulares.
- Listas ortogonales.

MAPA CONCEPTUAL

ESTRUCTURAS DE DATOS LINEALES

Cada componente tiene un único sucesor.

Y un único predecesor con excepción del último y el primero.



REFERENCIAS BIBLIOGRAFICAS

- [1] Joyanes Aguilar, L. (2003). Fundamentos de programación: algoritmos y estructura de datos y objetos.

- [2] Loomis, M. E. (1991). Estructura de datos y organización de archivos. Prentice-Hall Hispanoamericana.

- [3] Yarleque Ferrer, R. E. (2018). ESTRUCTURA DE DATOS Introducción, Conceptos, tipos de datos, clasificación general, Arrays, listas enlazadas, Pilas, Colas, Inicialización y asignación de valores.

- [4] Sisa, A. J., & Muñoz, J. V. (2002). Estructuras de datos y algoritmos: con énfasis en programación orientada a objetos. Pearson Educación de Colombia.

- [5] Manuel, L. M. V. Unidad II. Estructuras de Datos Lineales-Listas.

- [6] Jorge, J. S. Estructuras de datos: Pilas, Colas, Listas.

- [7] Moltó Martínez, G. (2011). Estructuras de Datos Lineales: Pila, Cola y Lista con Punto de Interés.