Institute of
Technology
Development
of Canada

475 Granville Street, Vancouver, BC, V6C 1T1
Phone: +1(604)558-8727, +1(604)409-8200
Toll Free: +1(888) 880-4410
Fax: +1(888) 881-6545
Web: www.itdcanada.ca
Email: studying@itdcanada.ca

Linked Lists In C

Institute of Technology Development of Canada

475 Granville Street, Vancouver, BC, V6C 1T1
Phone: +1(604)558-8727, +1(604)409-8200
Toll Free: +1(888) 880-4410
Fax: +1(888) 881-6545
Web: www.itdcanada.ca
Email: studying@itdcanada.ca

What is a Linked List?

- A linked list is a data structure that consists of a sequence of nodes.
- Each node contains data and a reference (or link) to the next node in the sequence.
- The last node points to NULL, indicating the end of the list.

Institute of Technology Development of Canada

475 Granville Street, Vancouver, BC, V6C 1T1
Phone: +1(604)558-8727, +1(604)409-8200
Toll Free: +1(888) 880-4410
Fax: +1(888) 881-6545
Web: www.itdcanada.ca
Email: studying@itdcanada.ca

Types of Linked Lists

- Singly Linked List: Each node has a data field and a next pointer.
- Doubly Linked List: Each node has a data field, a next pointer, and a previous pointer.
- Circular Linked List: Last node points back to the first node.

Institute of
Technology
Development
of Canada

475 Granville Street, Vancouver, BC, V6C 1T1
Phone: +1(604)558-8727, +1(604)409-8200
Toll Free: +1(888) 880-4410
Fax: +1(888) 881-6545
Web: www.itdcanada.ca
Email: studying@itdcanada.ca

Linked List Operations

Institute of
Technology
Development
of Canada

475 Granville Street, Vancouver, BC, V6C 1T1
Phone: +1(604)558-8727, +1(604)409-8200
Toll Free: +1(888) 880-4410
Fax: +1(888) 881-6545
Web: www.itdcanada.ca
Email: studying@itdcanada.ca

Insertion

- Inserting at the beginning of the list.
- Inserting at the end of the list.
- Inserting at a specific position.

Institute of Technology Development of Canada

475 Granville Street, Vancouver, BC, V6C 1T1
Phone: +1(604)558-8727, +1(604)409-8200
Toll Free: +1(888) 880-4410
Fax: +1(888) 881-6545
Web: www.itdcanada.ca
Email: studying@itdcanada.ca

Deletion

- Deleting from the beginning of the list.
- Deleting from the end of the list.
- Deleting a specific node.

Institute of
Technology
Development
of Canada

475 Granville Street, Vancouver, BC, V6C 1T1
Phone: +1(604)558-8727, +1(604)409-8200
Toll Free: +1(888) 880-4410
Fax: +1(888) 881-6545
Web: www.itdcanada.ca
Email: studying@itdcanada.ca

Traversal

- Iterating through the linked list to access each node and its data.

Institute of Technology Development of Canada

475 Granville Street, Vancouver, BC, V6C 1T1
Phone: +1(604)558-8727, +1(604)409-8200
Toll Free: +1(888) 880-4410
Fax: +1(888) 881-6545
Web: www.itdcanada.ca
Email: studying@itdcanada.ca

Advantages of Linked Lists

- Dynamic size: Linked lists can grow or shrink as needed.
- Efficient insertion and deletion: Adding or removing elements can be done easily by adjusting the pointers.
- No preallocation of memory: Memory is allocated on-demand.

Institute of Technology Development of Canada

475 Granville Street, Vancouver, BC, V6C 1T1
Phone: +1(604)558-8727, +1(604)409-8200
Toll Free: +1(888) 880-4410
Fax: +1(888) 881-6545
Web: www.itdcanada.ca
Email: studying@itdcanada.ca

Disadvantages of Linked Lists

- Sequential access: Random access is not possible, and elements must be accessed sequentially.
- Extra memory: Linked lists require additional memory for storing the next pointers.

Institute of Technology Development of Canada

475 Granville Street, Vancouver, BC, V6C 1T1
Phone: +1(604)558-8727, +1(604)409-8200
Toll Free: +1(888) 880-4410
Fax: +1(888) 881-6545
Web: www.itdcanada.ca
Email: studying@itdcanada.ca

Advantages of Linked Lists

- Defining a structure for the linked list node.
- Initializing the head pointer.
- Implementing the various operations using pointers and memory allocation.

Institute of Technology Development of Canada

475 Granville Street, Vancouver, BC, V6C 1T1
Phone: +1(604)558-8727, +1(604)409-8200
Toll Free: +1(888) 880-4410
Fax: +1(888) 881-6545
Web: www.itdcanada.ca
Email: studying@itdcanada.ca

Implementation of Linked List

- Defining a structure for the linked list node.
- Initializing the head pointer.
- Implementing the various operations using pointers and memory allocation.

Institute of Technology Development of Canada

475 Granville Street, Vancouver, BC, V6C 1T1
Phone: +1(604)558-8727, +1(604)409-8200
Toll Free: +1(888) 880-4410
Fax: +1(888) 881-6545
Web: www.itdcanada.ca
Email: studying@itdcanada.ca

Use Cases

- Operating Systems:
- Process Management: Linked lists are used to manage the list of active processes, scheduling queues, and maintaining process control blocks.
- File Systems: Linked lists are utilized to maintain the file allocation table (FAT) or inode structure, representing the organization of files on storage devices.
- Data Structures:
- Stacks and Queues: Linked lists are often used to implement stacks and queues, where nodes are added or removed from one end.
- Hash Tables: Some implementations of hash tables use linked lists to handle collisions, where multiple keys map to the same hash value.

Institute of Technology Development of Canada

475 Granville Street, Vancouver, BC, V6C 1T1
Phone: +1(604)558-8727, +1(604)409-8200
Toll Free: +1(888) 880-4410
Fax: +1(888) 881-6545
Web: www.itdcanada.ca
Email: studying@itdcanada.ca

Use Cases

- Network Data Structures:
- Routing Tables: In networking, linked lists are used to store routing information for packet forwarding, allowing efficient updates and lookups.
- Adjacency Lists: Linked lists can represent adjacency lists for graphs, used in graph algorithms and network analysis.
- Language Implementations:
- Garbage Collection: Some garbage collection algorithms, like the Mark-Sweep algorithm, employ linked lists to track and manage memory allocations.
- Interpreter and Compiler Structures: Linked lists are used in various language structures, such as symbol tables, abstract syntax trees, and linked representation of code constructs.
- Implementing Higher-Level Data Structures:
- Linked List Variants: Doubly linked lists and circular linked lists are often used as building blocks for more complex data structures like linked queues, linked stacks, and linked deques.
- Linked Trees: In some tree structures, such as threaded binary trees or self-balancing trees like AVL or Red-Black trees, linked lists can be used to maintain relationships between nodes.

Institute of Technology Development of Canada

475 Granville Street, Vancouver, BC, V6C 1T1
Phone: +1(604)558-8727, +1(604)409-8200
Toll Free: +1(888) 880-4410
Fax: +1(888) 881-6545
Web: www.itdcanada.ca
Email: studying@itdcanada.ca

Use Cases

- Network Data Structures:
- Routing Tables: In networking, linked lists are used to store routing information for packet forwarding, allowing efficient updates and lookups.
- Adjacency Lists: Linked lists can represent adjacency lists for graphs, used in graph algorithms and network analysis.
- Language Implementations:
- Garbage Collection: Some garbage collection algorithms, like the Mark-Sweep algorithm, employ linked lists to track and manage memory allocations.
- Interpreter and Compiler Structures: Linked lists are used in various language structures, such as symbol tables, abstract syntax trees, and linked representation of code constructs.
- Implementing Higher-Level Data Structures:
- Linked List Variants: Doubly linked lists and circular linked lists are often used as building blocks for more complex data structures like linked queues, linked stacks, and linked deques.
- Linked Trees: In some tree structures, such as threaded binary trees or self-balancing trees like AVL or Red-Black trees, linked lists can be used to maintain relationships between nodes.

Institute of Technology Development of Canada

475 Granville Street, Vancouver, BC, V6C 1T1
Phone: +1(604)558-8727, +1(604)409-8200
Toll Free: +1(888) 880-4410
Fax: +1(888) 881-6545
Web: www.itdcanada.ca
Email: studying@itdcanada.ca

Use Cases

- Animation Sequencing: In games, animations are often a sequence of frames. A linked list can be used to store these frames, where each node represents a frame. The 'next' pointer can be used to traverse to the next frame in the sequence, making it easy to loop through animations.
- Pathfinding: In games where characters need to navigate through a map, linked lists can be used to store the path that a character needs to follow. Each node in the list could represent a waypoint or a step in the path.
- Inventory Systems: In RPGs or adventure games, players often have an inventory of items. A linked list can be used to manage this inventory, where each node represents an item. This allows for easy addition and removal of items.
- AI Behavior Trees: In games with AI characters, a linked list can be used to represent a behavior tree, where each node represents a behavior or decision point. This allows the AI to traverse through different behaviors based on game conditions.

Institute of Technology Development of Canada

475 Granville Street, Vancouver, BC, V6C 1T1
Phone: +1(604)558-8727, +1(604)409-8200
Toll Free: +1(888) 880-4410
Fax: +1(888) 881-6545
Web: www.itdcanada.ca
Email: studying@itdcanada.ca

Use Cases

- Undo/Redo Functionality: In games like chess or other strategy games, an action history can be maintained using a linked list, where each node represents a game state. This allows for an easy implementation of undo/redo functionality.
- Particle Effects: In games with particle effects (like explosions, magic effects, etc.), each particle can be a node in a linked list. This allows for easy creation, updating, and deletion of particles.
- Memory Management: Linked lists can be used to manage memory in games, especially in consoles where memory is limited. They can be used to implement custom memory allocators that are more efficient for the specific needs of the game.