

0.1 1333203 DSA Winter 2023

0.1.1 Q1a: Define linked list. List different types of linked list. (03 marks)

Ans 1a: A linked list is a dynamic data structure consisting of a sequence of elements, where each element (called a node) contains data and a reference (or link) to the next element in the sequence. Unlike arrays, linked lists do not store elements in contiguous memory locations, allowing for efficient insertion and deletion operations.

Key characteristics of linked lists:

- Dynamic size: Can grow or shrink during program execution
- Non-contiguous memory allocation: Elements can be stored anywhere in memory
- Efficient insertion and deletion: $O(1)$ time complexity for operations at the beginning or end

Different types of linked lists:

1. Singly Linked List:

- Each node contains data and a single reference to the next node
- Last node points to NULL, indicating the end of the list

2. Doubly Linked List:

- Each node contains data and two references: one to the next node and one to the previous node
- Allows traversal in both directions

3. Circular Linked List:

- Similar to singly linked list, but the last node points back to the first node
- Forms a closed loop

4. Circular Doubly Linked List:

- Combines features of doubly linked and circular linked lists
- Last node's next pointer points to the first node, and first node's previous pointer points to the last node

5. Header Linked List:

- Contains a special header node at the beginning
- Header node may store metadata about the list (e.g., size, pointers to first and last elements)

0.1.2 પ્રશ્ન 1અ: લીન્કડ લીસ્ટની વ્યાખ્યા આપો. વવિધિ પ્રકારના લીન્કડ લીસ્ટ ની યાદી આપો. (૦૩ ગુણ)

જવાબ 1અ: linked list એ એક ડાયનેમિક ડેટા સ્ટ્રક્ચર છે જેમાં એલમિન્ટ્સનો ક્રમ હોય છે, જ્યાં દરેક એલમિન્ટ (જેને node કહેવાય છે) ડેટા અને ક્રમમાં આગળના એલમિન્ટનો સંદર્ભ (અથવા link) ધરાવે છે. એરેઝથી વધીરત, linked lists એલમિન્ટ્સને સતત મેમરી સ્થાનોમાં સંગ્રહિત કરતા નથી, જે insertion અને deletion ઓપરેશન્સને કાર્યક્ષમ બનાવે છે.

linked lists ની મુખ્ય લાક્ષણિકતાઓ:

- ડાયનેમિક કદ: પ્રોગ્રામ એક્ઝીક્યુશન દરમિયાન વધી અથવા ઘટી શકે છે
- નોન-કન્ટીગ્યુઅસ મેમરી એલોકેશન: એલમિન્ટ્સ મેમરીમાં ગમે ત્યાં સ્ટોર કરી શકાય છે
- કાર્યક્ષમ insertion અને deletion: શરૂઆત અથવા અંતમાં ઓપરેશન્સ માટે $O(1)$ સમય જટિલતા

વવિધિ પ્રકારના linked lists:

1. Singly Linked List:

- દરેક node ડેટા અને આગળના node નો એક સિંગલ સંદર્ભ ધરાવે છે
- છેલ્લું node NULL તરફ પોઈન્ટ કરે છે, જે લિસ્ટના અંતને સૂચવે છે

2. Doubly Linked List:

- દરેક node ડેટા અને બે સંદર્ભો ધરાવે છે: એક આગળના node માટે અને એક પાછલા node માટે
- બંને દિશાઓમાં ટ્રાવર્સલની મંજૂરી આપે છે

3. Circular Linked List:

- Singly Linked List જેવું જ, પરંતુ છેલ્લું node પ્રથમ node તરફ પાછું પોઈન્ટ કરે છે
- બંધ લૂપ બનાવે છે

4. Circular Doubly Linked List:

- Doubly Linked અને Circular Linked Lists ની વર્ણિતતાઓને જોડે છે
- છેલ્લા node નો next pointer પ્રથમ node તરફ પોઈન્ટ કરે છે, અને પ્રથમ node નો previous pointer છેલ્લા node તરફ પોઈન્ટ કરે છે

5. Header Linked List:

- શરૂઆતમાં એક વર્ણિત header node ધરાવે છે
- Header node લિસ્ટ વર્ણિત મેટાડેટા સ્ટોર કરી શકે છે (દા.ત., કદ, પ્રથમ અને છેલ્લા એલમિન્ટ્સના pointers)