course_content

top

Data Structures and Algorithms

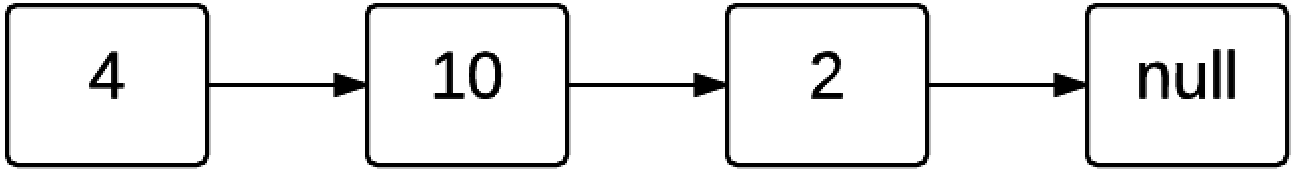
Linked Lists

Week 5

Time:
5 min

Linked Lists

- Data is held in **nodes**
- Each node holds information and a pointer to the next node (null for last one)
- Efficient deleting and inserting (if you have a reference to node)
- (



Source: Brilliant

Time:
5 min

Linked Lists

- The fundamental objects are **nodes** (a class that you will have to create)
- Each node object will have a **getValue** and a **getNext** method that takes you to the following node (The null node returns null for each)
- Each node also has a **setValue** and **setNext** method to alter these.

- A **linked list** object is simply a pointer to the **head** node.
- Some linked lists also include a reference to the **tail** node

Time:
5 min

Linked List - Primary Methods

- List() - creates an empty Linked List
- head() - returns the head node
- popFirst() - 2nd node becomes head
- prepend(val) - val becomes head and current head becomes 2nd node

- tail() - returns the tail node **(if implemented)**
- pop() - 2nd to last node becomes tail
- append(val) - val becomes tail and current tail becomes 2nd to last node

Time:
15 min

Linked List - PseudoCode Exercise

- Groups of 3, no internet.
- Write pseudo-code for the following functions in the case of linked-lists with tail references
    - Access i’th element
    - Search for an element (e.g. “hello”)
    - Insert a node (assuming you know the node before the space to insert)
    - Delete a given node (assuming you know the previous node)

Time:
5 min

Linked Lists - Think, Pair, Share

- What are the benefits of linked lists?
- What are the disadvantages of linked lists?
- When should we use them? Think about:
    - Memory usage
    - Dynamic lists/arrays
    - Time complexity for basic methods

Time:
5 min

Search and Share - Doubly-linked Lists

- What are doubly-linked lists?
    - Where might the doubled link come from?
    - What value is there in a double link?
    - What are the costs of a double link?
    - Do any operations get easier/harder with doubled-links?
- Extra implementation options
    - Circular Linked Lists (no Null node - start/end)
    - Sentinel Nodes (Dummy header/trailer nodes)

Time:
5 min

Linked Lists - Applications

- Separate Chaining in Hash Tables (implemented as linked lists)
- Used for queues and stacks when you don’t need to access the middle
- When you don’t have **‘contiguous’** memory
- Circular linked lists are used in operating systems to give a fixed time slot for running and looping through processes one-by-one.

Time:
1 min

Questions