

CE205 Data Structures Week-4

Tree Data Structure Types and Applications (Binary Tree, Tree Traversals, Heaps)

Author: Asst. Prof. Dr. Uğur CORUH

Contents

1	CE205 Data Structures	1
2	Week-4	1
2.0.1	Tree Data Structure Types and Applications (Binary Tree, Tree Traversals, Heaps) . .	1
2.0.2	Outline	2
2.0.3	Graph Representation Tools	2
2.0.4	Tree - Terminology	2
2.0.5	Tree Representations	2
2.0.6	Binary Tree Datastructure	2
2.0.7	Longest Common Ancestor	3
2.0.8	Longest Common Ancestor	3
2.0.9	Binary Tree Representations	3
2.0.10	Binary Tree Traversals	4
2.0.11	Threaded Binary Trees	4
2.0.12	Max Priority Queue	4
2.0.13	Heap Data Structure	4
2.0.14	Heap Data Structure	4
2.0.15	Heap Data Structure	4
2.0.16	Heap Data Structure	5
2.0.17	Heap Data Structure	5
2.0.18	Heap Data Structure	5
2.0.19	Heap Data Structure	5
2.0.20	Heap Data Structure	5
2.0.21	Heap Data Structure	5

List of Figures

List of Tables

1 CE205 Data Structures

2 Week-4

2.0.1 Tree Data Structure Types and Applications (Binary Tree, Tree Traversals, Heaps)

Download DOC¹, SLIDE², PPTX³

¹[ce205-week-4-tree-structures.md_doc.pdf](#)

²[ce205-week-4-tree-structures.md_slide.pdf](#)

³[ce205-week-4-tree-structures.md_slide.pptx](#)

2.0.2 Outline

- Graph Representation Tools
 - Tree Structures and Binary Tree and Traversals (In-Order, Pre-Order, Post-Order)
 - Heaps (Max, Min, Binary , Binomial, Fibonacci, Leftist, K-ary) and Priority Queue
 - Heap Sort
 - Huffman Coding
-

2.0.3 Graph Representation Tools

- Microsoft Automatic Graph Layout
 - <https://www.microsoft.com/en-us/download/details.aspx?id=52034>
 - <https://github.com/microsoft/automatic-graph-layout>
 - Graphviz
 - <https://graphviz.org/resources/>
 - Plantuml
 - https://ucoruh.github.io/ce204-object-oriented-programming/week-5/ce204-week-5/#calling-plantuml-from-java_1
-

2.0.4 Tree - Terminology

- Btech Smart Class
 - http://www.btechsmartclass.com/data_structures/tree-terminology.html
-

2.0.5 Tree Representations

- Btech Smart Class
 - http://www.btechsmartclass.com/data_structures/tree-representations.html
-

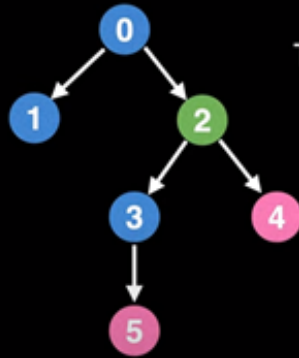
2.0.6 Binary Tree Datastructure

1. Construction and Conversion
 2. Checking and Printing
 3. Summation
 4. Longest Common Ancestor
- Btech Smart Class
 - http://www.btechsmartclass.com/data_structures/binary-tree.html
 - William Fiset
 - https://www.youtube.com/watch?v=sD1IoalFomA&ab_channel=WilliamFiset
-

2.0.7 Longest Common Ancestor

Definition

The **Lowest Common Ancestor** (LCA) of two nodes 'a' and 'b' in a **rooted tree** is the deepest node 'c' that has both 'a' and 'b' as descendants (where a node can be a descendant of itself)



$LCA(5, 4) = 2$

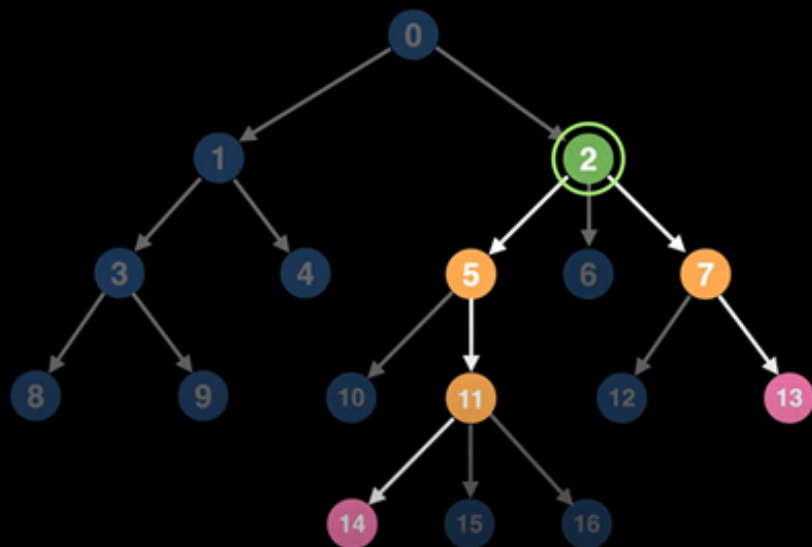
The LCA problem has several applications in Computer Science, notably:

- Finding the distance between two nodes
- Inheritance hierarchies in OOP
- As a subroutine in several advanced algorithms and data structures
- etc...

2.0.8 Longest Common Ancestor

Understanding LCA

$LCA(13, 14) = 2$



2.0.9 Binary Tree Representations

- Btech Smart Class

- http://www.btechsmartclass.com/data_structures/binary-tree-representations.html
-

2.0.10 Binary Tree Traversals

- Btech Smart Class
 - http://www.btechsmartclass.com/data_structures/binary-tree-traversals.html
 - * In-Order
 - * Pre-Order
 - * Post-Order
-

2.0.11 Threaded Binary Trees

- Btech Smart Class
 - http://www.btechsmartclass.com/data_structures/threaded-binary-trees.html
-

2.0.12 Max Priority Queue

- Btech Smart Class
 - http://www.btechsmartclass.com/data_structures/max-priority-queue.html
 - William Fiset
 - https://www.youtube.com/watch?v=wpTEvk0bshY&t=0s&ab_channel=WilliamFiset
 - <https://github.com/williamfiset/Algorithms/tree/master/src/main/java/com/williamfiset/algorithms/datastructures>
-

2.0.13 Heap Data Structure

- Heap Sort
 - <https://ucoruh.github.io/ce100-algorithms-and-programming-II/week-4/ce100-week-4-heap/>
-

2.0.14 Heap Data Structure

- Programiz
 - <https://www.programiz.com/dsa/heap-data-structure>
 - Btech Smart Class
 - Max-Heap
 - * http://www.btechsmartclass.com/data_structures/max-heap.html
 - Geeks for Geeks
 - Binary Heap
 - * <https://www.geeksforgeeks.org/binary-heap/?ref=lbp>
 - * <https://www.geeksforgeeks.org/difference-between-binary-heap-binomial-heap-and-fibonacci-heap/?ref=rp>
-

2.0.15 Heap Data Structure

- Binomial Heap
 - Geeks for Geeks
 - * <https://www.geeksforgeeks.org/binomial-heap-2/>
-

2.0.16 Heap Data Structure

1. Structure of Fibonacci Heaps
2. Mergeable-heap operations

- Fibonacci Heap
 - William Fiset
 - * <https://github.com/williamfiset/Algorithms/tree/master/src/main/java/com/williamfiset/algorithms/datastructures/heap/FibonacciHeap.java>
 - Geeks for Geeks
 - * <https://www.geeksforgeeks.org/fibonacci-heap-set-1-introduction/?ref=lbp>
-

2.0.17 Heap Data Structure

1. Decreasing a key and deleting a node
2. Bounding the maximum degree

- Heap Operations
 - <https://www.geeksforgeeks.org/fibonacci-heap-insertion-and-union/?ref=lbp>
 - <https://www.geeksforgeeks.org/fibonacci-heap-deletion-extract-min-and-decrease-key/?ref=lbp>
-

2.0.18 Heap Data Structure

- Leftist Heap
 - Geeks for Geeks
 - * <https://www.geeksforgeeks.org/leftist-tree-leftist-heap/?ref=lbp>
 - Toronto
 - * https://www.dgp.toronto.edu/public_user/JamesStewart/378notes/10leftist/
-

2.0.19 Heap Data Structure

- Geeks for Geeks
 - <https://www.geeksforgeeks.org/k-ary-heap/?ref=lbp>
-

2.0.20 Heap Data Structure

- Heap Sort
 - <https://ucoruh.github.io/ce100-algorithms-and-programming-II/week-4/ce100-week-4-heap/>
-

2.0.21 Heap Data Structure

- Huffman Coding
 - Geeks for Geeks
 - * <https://www.geeksforgeeks.org/difference-between-binary-heap-binomial-heap-and-fibonacci-heap/?ref=rp>
-

End – Of – Week – 4