

EC504 Project Proposal:

Tree Algorithms and Applications

Ziyan Chen

Hin Lui Shum

Jia Wilkins

Zhiqing Wang

Project Objectives

Main Objective:

Analyze different tree structures for enhanced data management and processing efficiency.

Methods:

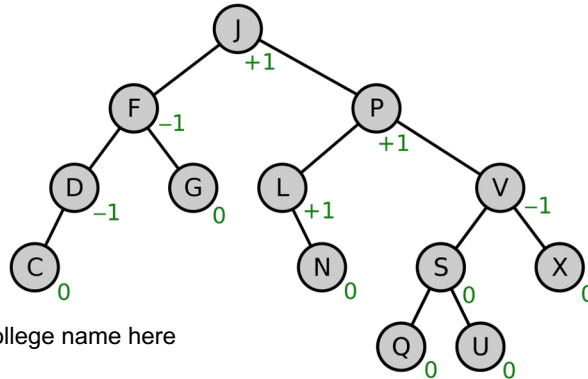
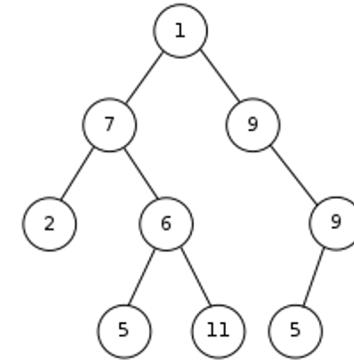
1. Analyze and application different types of trees.
2. Code Design and Implementation
3. Simulation and Visualization

Introduction to Tree Structures

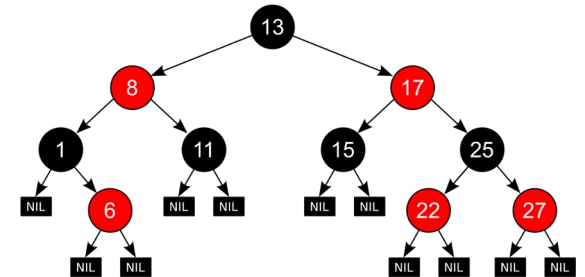
- Start from what we've learned:
 - Binary Tree, Balanced Tree, AVL Tree
 - Red-Black Tree
 - B-Tree
 - Monte Carlo Tree Search
- When there are more than 2 children:
 - KD-Tree
 - Special cases: Quad-Tree, Oct-Tree, (and probably 2^n Tree)
- When things go to linear:
 - Fusion Tree
 - Van Emde Boas Tree

Binary Trees and Derivatives

- Binary Tree
 - Basic structure and properties.
- Balanced Tree
 - Definition and significance.
- AVL Tree
 - How it maintains balance; use cases.
- Red-Black Tree
 - Unique properties; comparison with AVL.

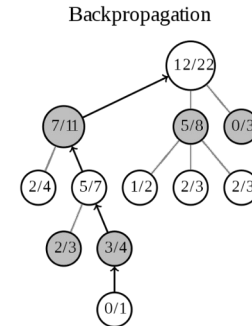
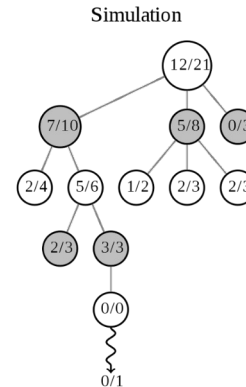
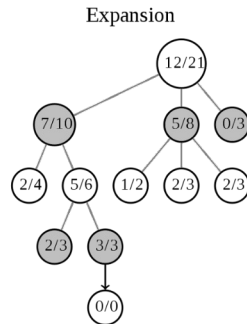
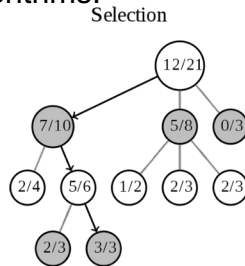
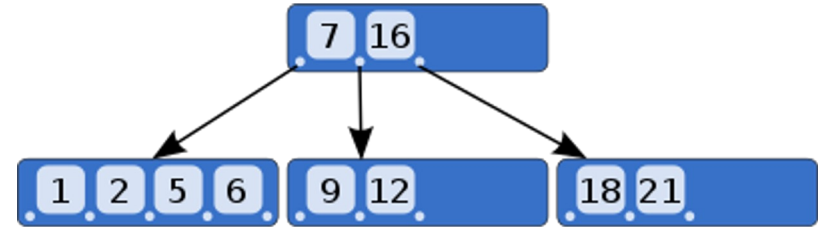


Boston University School/college name here



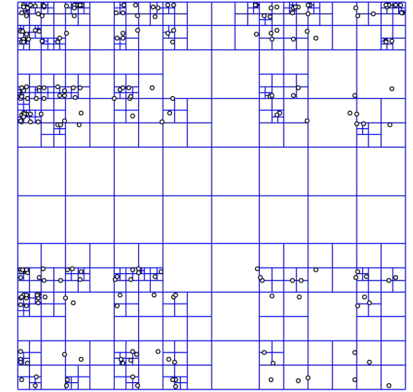
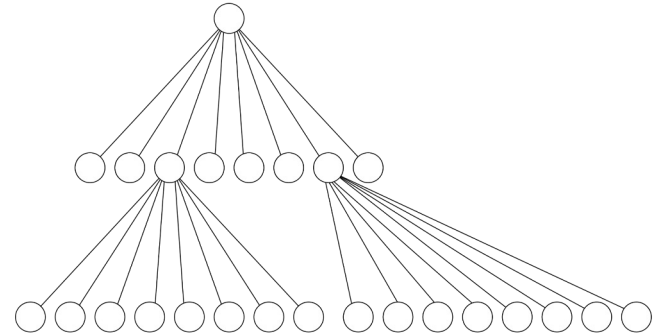
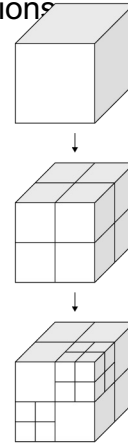
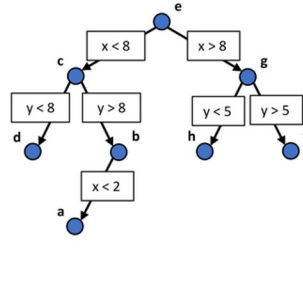
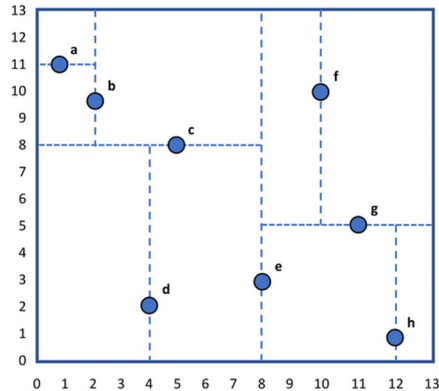
Advanced Tree Structures

- B-Tree
 - Structure, properties, and application in databases.
- Monte Carlo Tree Search
 - Explanation and application in AI, particularly in games and decision-making algorithms.



Trees with Multiple Children

- KD-Tree
 - Structure and application in multi-dimensional search spaces.
- Special Cases
 - Quad-Tree: Usage in spatial indexing, image processing.
 - Oct-Tree: Application in 3D space partitioning.
 - 2^n Tree: Generalization and potential applications



Linear Tree Structures

- Fusion Tree
 - Description and use in efficient searching.
- Van Emde Boas Tree
 - Structure and application in priority queues and network optimization.

