

# Comp 352 Winter 2019

## Tutorial 7

March 6, 2019



# Outline

1. Binary Search Tree

2. AVL Tree



# Announcement

1. None



# Binary Search Tree - Definition

- tree
- binary
- proper (heap?)
- external nodes do not store values
- $key(v_{left}) \leq key(v) \leq key(v_{right})$ .

# Binary Search Tree - Properties

- inorder traversal
- $BST \in heap$  or  $heap \in BST$ ?

# Binary Search Tree - Operations and Performance

- Search
- Insertion
- Removal: inorder traversal
  - At most one internal node
  - Two internal nodes
    - Leftmost
    - Rightmost
- $O(h)$

# Balanced Search Tree

- Rotation
- Restructure
- $a, b, c$  and  $x, y, z$

## AVL Tree - Definition

- Inventors: Adelson-Velskii and Landis
- binary search tree
- for every internal node  $v$  of  $T$ :  $|h(v_{left}) - h(v_{right})| \leq 1$
- balanced



## AVL Tree - Properties

- The height of an AVL tree storing  $n$  keys is  $O(\log n)$ .

# AVL Tree - Operations and Performance

- Search
- Insertion
  - Trinode Restructuring
  - Single rotation
  - Double rotation
- Removal