COP 3540 Data Structures with OOP

Chapter 8 - Part 1
Binary Trees

Why Trees?

- Trees are one of the fundamental data structures.
- Many real-world phenomena cannot be represented w/data structures we've had so far.

- Think of arrays:

  - Easy to **search**, especially if ordered.
    - $O(\log_2 n)$ performance! Great. (binary search)

  - **Inserting; Deleting**? Horrible if ordered. Must find item or place before actions.

Why Trees?

- How about Linked Lists?
  - Inserts and deletes? Great. Take O(1) time – the best you can get! (if inserting / deleting from one end)
  - Searching? Search to insert / delete/ change? Not nearly as good as O(1) or even O($\log_2 n$)!
    - On average, must search n/2 items!
    - Process requires O(n) time.
    - Ordering the linked list may help, as we must still search to find.