Discussion 07

Trees, Linked Lists

Aditya Balasubramanian
`aditbala [at] berkeley [dot] edu`

Announcements 📣

- Cats due today
- Ants released tommorow
  - Checkpoint due Friday
- Magic: the Lambda-ing will be released as an extra credit project
  - Smaller project
- HW04, Lab07, due Thursday
- Clobber policy
  - If you do better on final, can clobber midterm
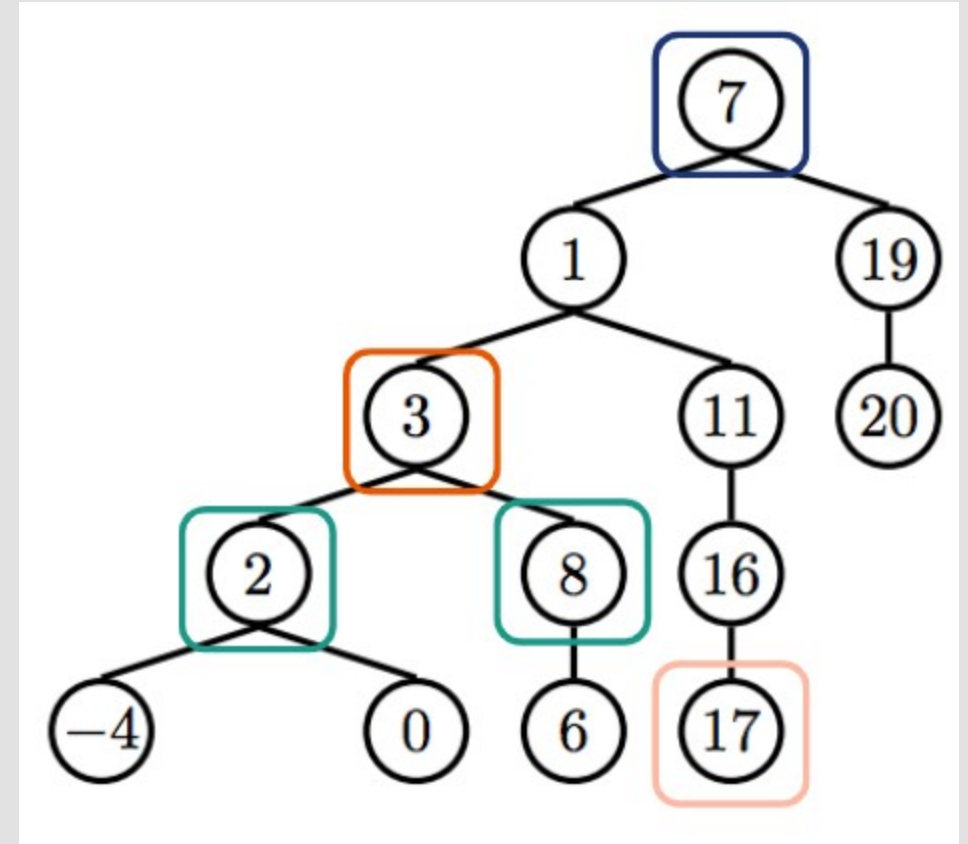- Midterm Discussion

Trees 🌳🌳🌳

Tree

- What are they?
  - Data structure for hierarchies of data
- What should we know
  - Recursion!
  - Every subtree is also a Tree

Tree Terminology

- Parent Node
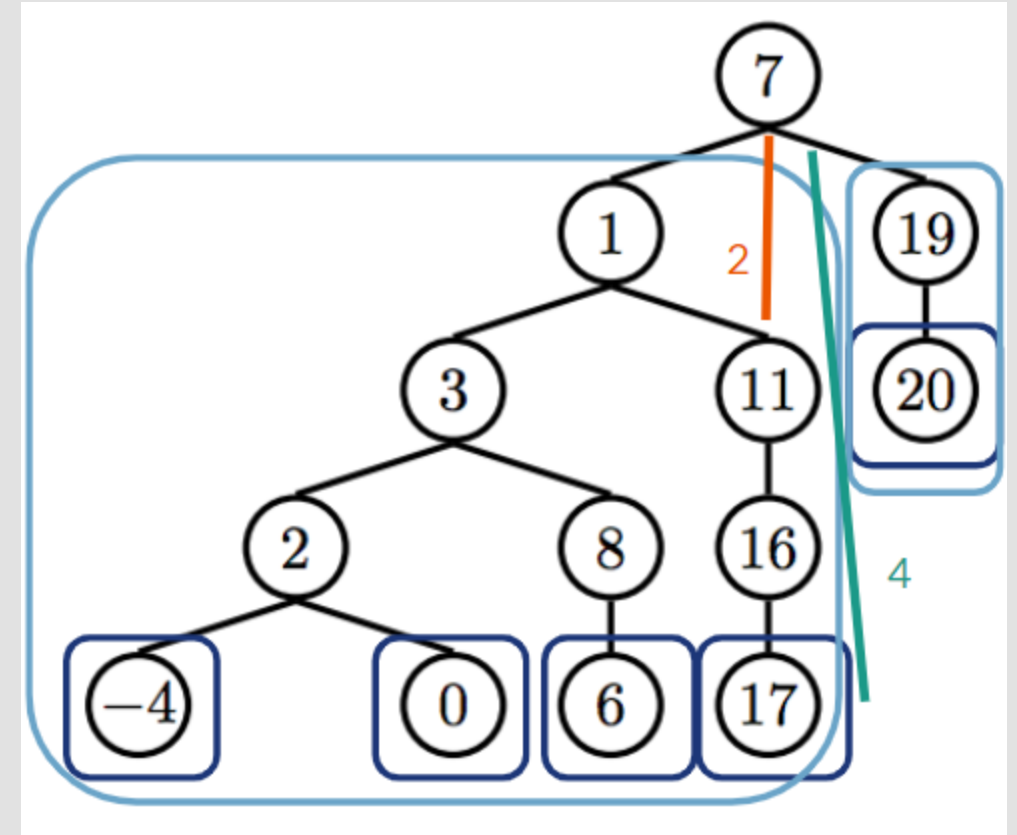 - A node that has branches
- Child Node
  - A node with a parent
  - Can only have one parent
- Root
  - The top node in a tree
  - There is only one root for a tree
- Label
  - The value of a node

More Tree Terminology

- Leaf
  - A node with no branches
- Branch
  - A subtree of the root
  - All branches are also trees
- Depth
  - How far away a node is from the root
- Height
  - The depth of the lowest leaf

Tree

- Initializing a `Tree`
  - `Tree(label, branches=[])`
  - `t = Tree(1, [Tree(2), Tree(3)])`
- Accessing branches of a `Tree`
  - `t.branches` `->` `[Tree(2), Tree(3)]`
- Checking if a `Tree` is a leaf
  - `t.is_leaf()` `->` `False`
  - `t.branches[0].is_leaf()` `->` `True`
- Getting label of a `Tree`
  - `t.label` `->` `1`

Manipulating Trees

`for b in t.branches`

- What is this?
  - IMPORTANT line for dealing with a `Tree`
- Why use this?
  - Allows us to iterate through branches of a `Tree`
  - Useful for calling recursive functions on all branches of a `Tree`
- Can also be a base case
  - The for loop does not run if there are no branches to iterate over

How do I use this? (Recursion for Trees)

1. Base Case
   - Smallest Input
   - Usually a leaf
2. Recursive Calls
   - Call recursive function on branches
3. Putting it together
   - Use recursive calls to solve problem
   - Can use `max`, `min`, `sum`, `any`, `all` on lists

Linked List 😳⏩😳⏩😳

Linked Lists

- Can be thought of as a queue waiting to enter a place (starting from the end of the line)
- Each person only knows themself and who is in front of them (rest of the line)

Linked Lists (In More Formal Terms)

- An object that either has a `first` and `rest` attribute or is empty
- `rest`
  - Must be `Link.empty` or another Linked List
  - Recursive data type!
- `first`
  - Any data type
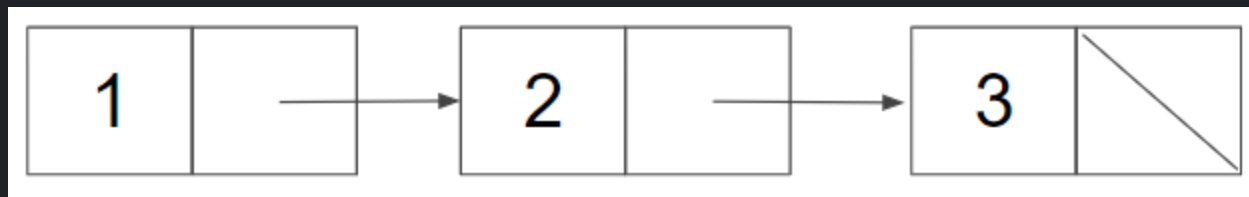
Linked List Example

```
one = Link(1, Link(2, Link(3)))
>>> one.first
1
>>> one.rest
Link(2, Link(3))
>>> one.rest.rest
Link(3)
>>> one.rest.rest.first
3
>>> Link(Link(1), Link(2, Link(3)))
>>> Link(1, 2)
```

Thank you!

Attendance Form -> https://tinyurl.com/adit-disc07

Anon Feedback -> https://tinyurl.com/adit-anon