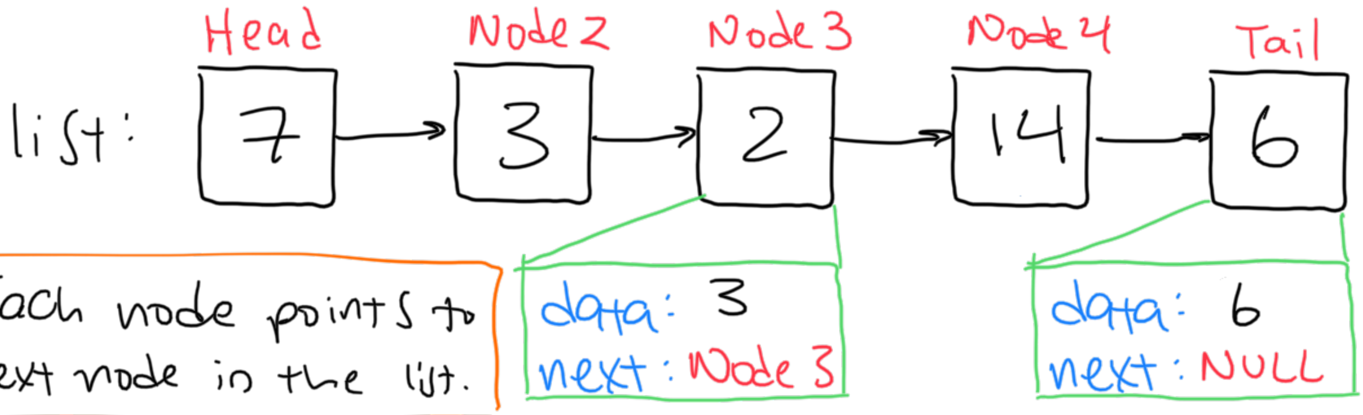
Linked Lists

James Bowden
Oct. 2020

What's a linked list?



Each node points to next node in the list.

How is it used?

list variable stores the head only:

```
Node list = head;          // 1st element. "Defines" the list.
Node curr = head.next;     // 2nd element
Node curr = curr.next;     // 3rd element
Node curr = curr.next;     // 4th element, and so on...
```

How would you add and remove elements?

When should I use it?

pros
- easily resizable
- efficient insertion/deletion
- plays nicely with recursion

cons
- slow search/sort
- takes more space
- not contiguous

Some modifications:

- Doubly linked lists – prev and next
  ↳ can go forward and backward
- Store head and tail
  ↳ can start at front or back
  ↳ easy to add/remove at front and back

What's it actually used for?

web browsers (back/fwd)

stacks

graphs/trees

queues

polynomial ops

heaps

music players

chronology (e.g. undo/redo func)

memory allocators :)