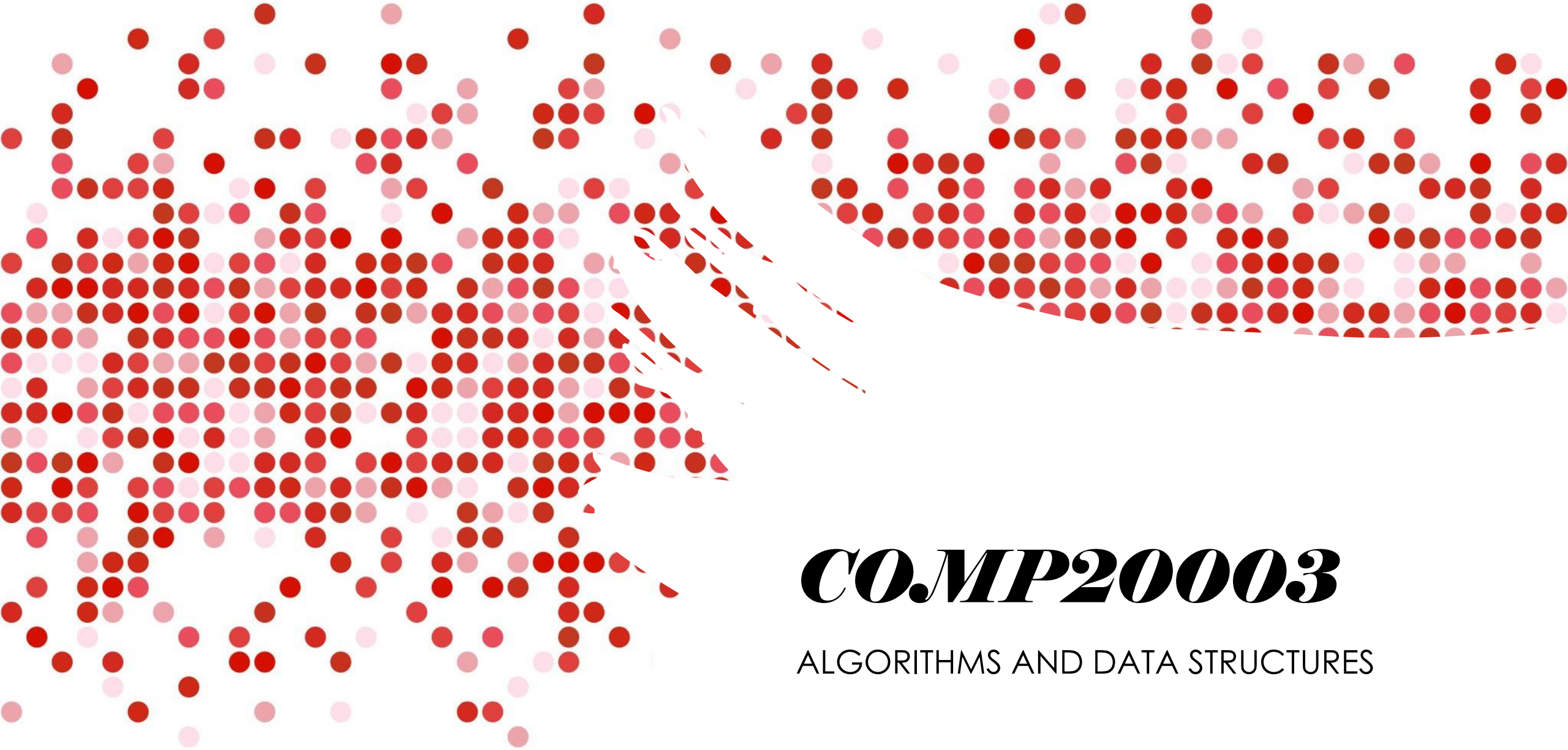


COMP20003

ALGORITHMS AND DATA STRUCTURES

Workshop

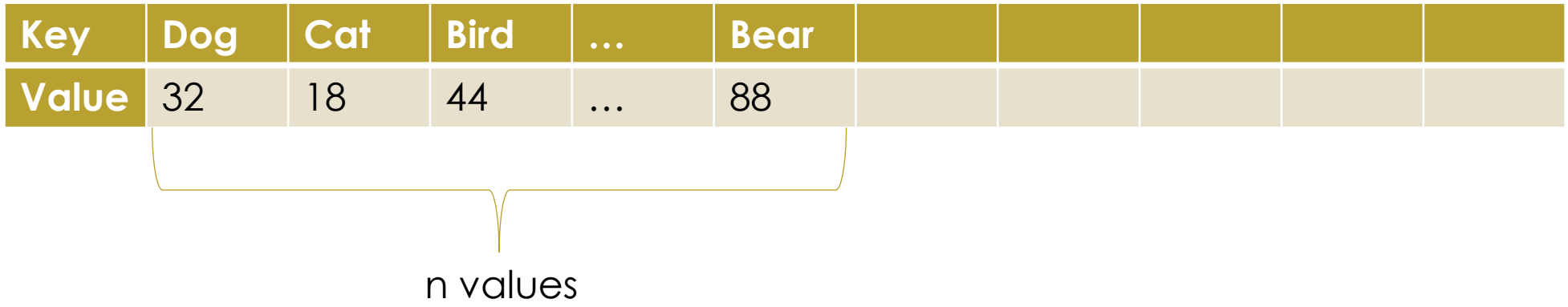
- Dictionary
- Data Structures
  - Static Arrays
  - Dynamic Arrays
  - Sorted Arrays
  - Linked Lists
  - Binary Search Trees

Dictionary

- Abstract Data Structure
- Search
  - Given a key, returns the value(s) for that key
- Insert
  - Given a key and value, stores those for later lookup
- Delete
  - Remove a given key/value pair

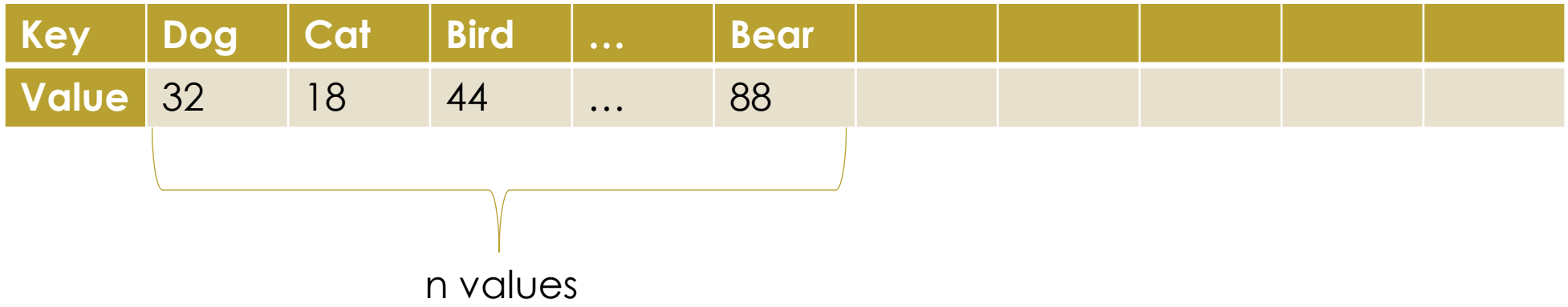
Static Arrays

- Insert
  - O(?)
- Search
  - O(?)



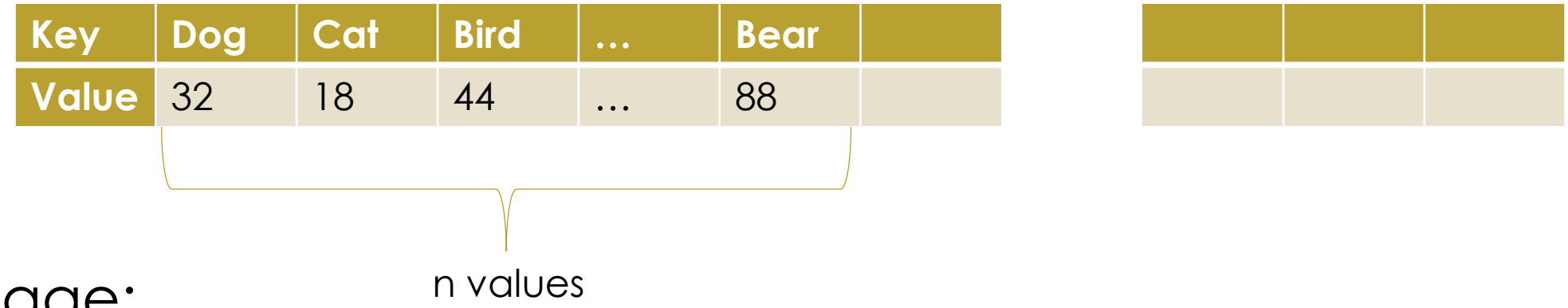
Static Arrays

- Insert
  - O(1)
- Search
  - O(n)



Dynamic Arrays

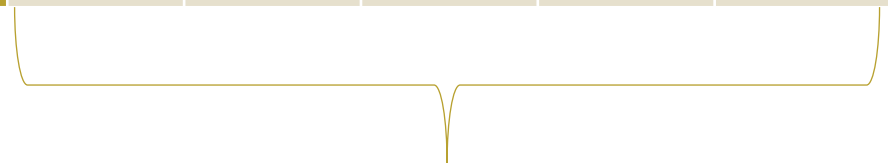
- Insert
  - O(?)
- Search
  - O(?)
- Advantage:
  - No limit to key/values & efficient space



Dynamic Arrays

- Insert
  - O(n)
- Search
  - O(n)
- Advantage:
  - No limit to key/values pairs & efficient space

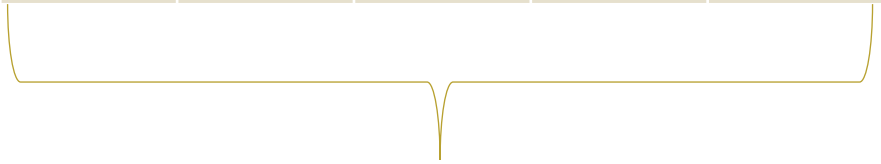
Key	Dog	Cat	Bird	…	Bear	
Value	32	18	44	…	88	



Sorted Dynamic Arrays

- Insert
  - O(?)
- Search
  - O(?)

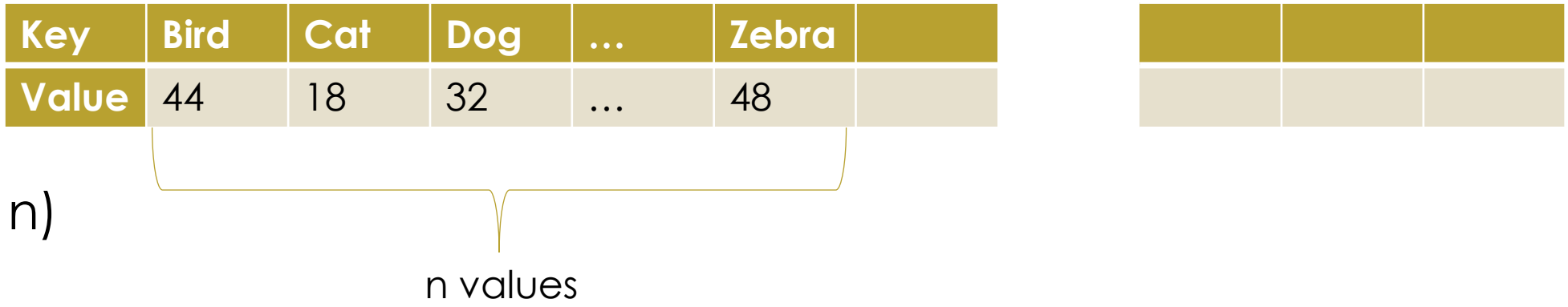
Key	Bird	Cat	Dog	…	Zebra	
Value	44	18	32	…	48	



n values

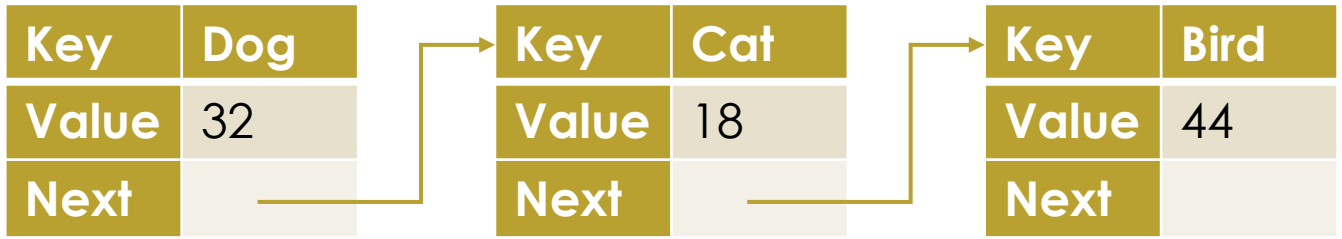
Sorted Dynamic Arrays

- Insert
  - O(n)
- Search
  - O(log n)



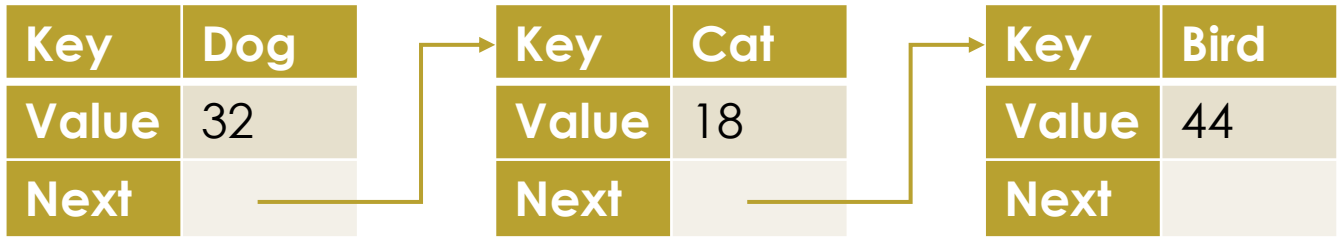
Linked List

- Insert
  - O(?)
- Search
  - O(?)
- Advantage:
  - No limit to key/value pairs & efficient space!



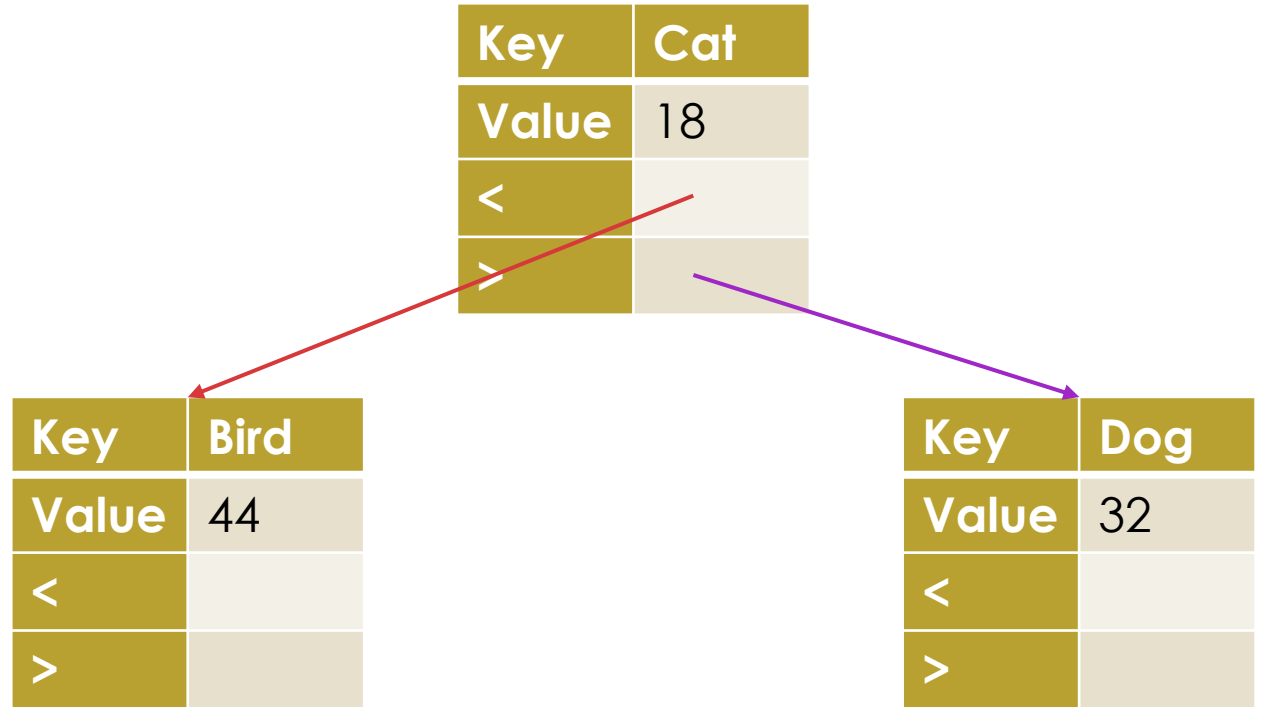
Linked List

- Insert
  - O(1)
- Search
  - O(n)
- Advantage:
  - No limit to key/value pairs & efficient space!



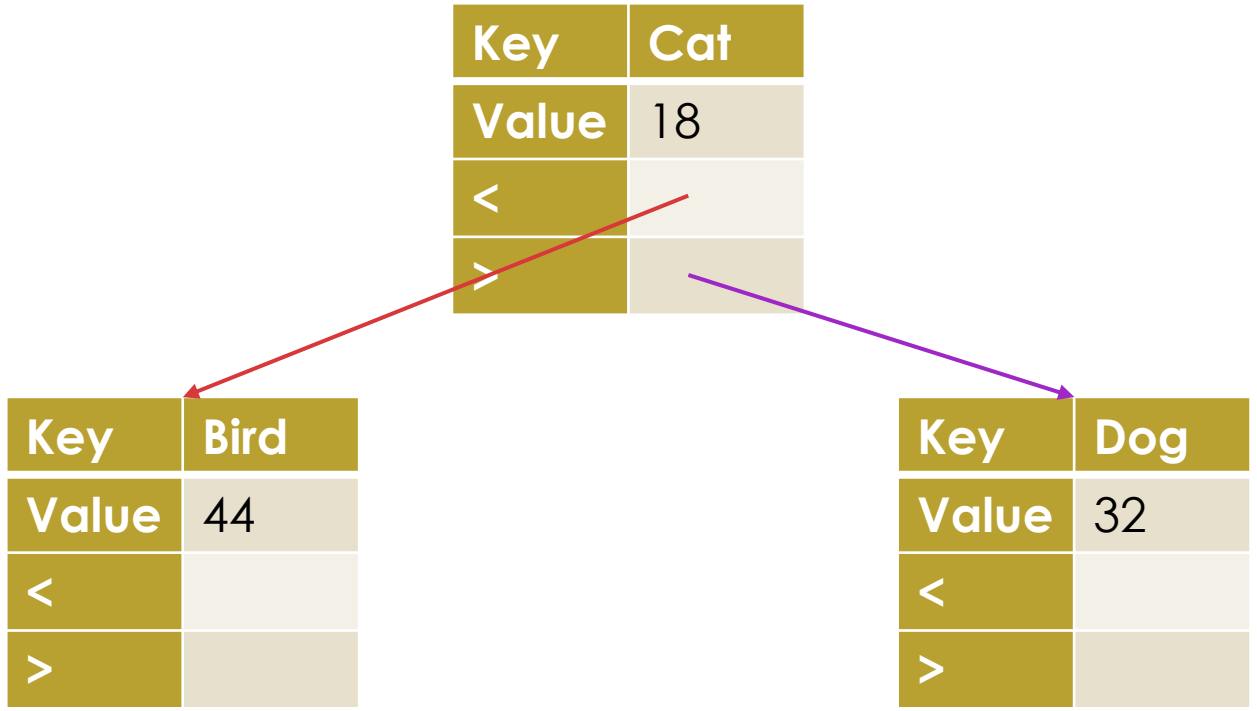
Binary Search Tree

- Binary Search built into list?
- Insert
  - O(?)
- Search
  - O(?)



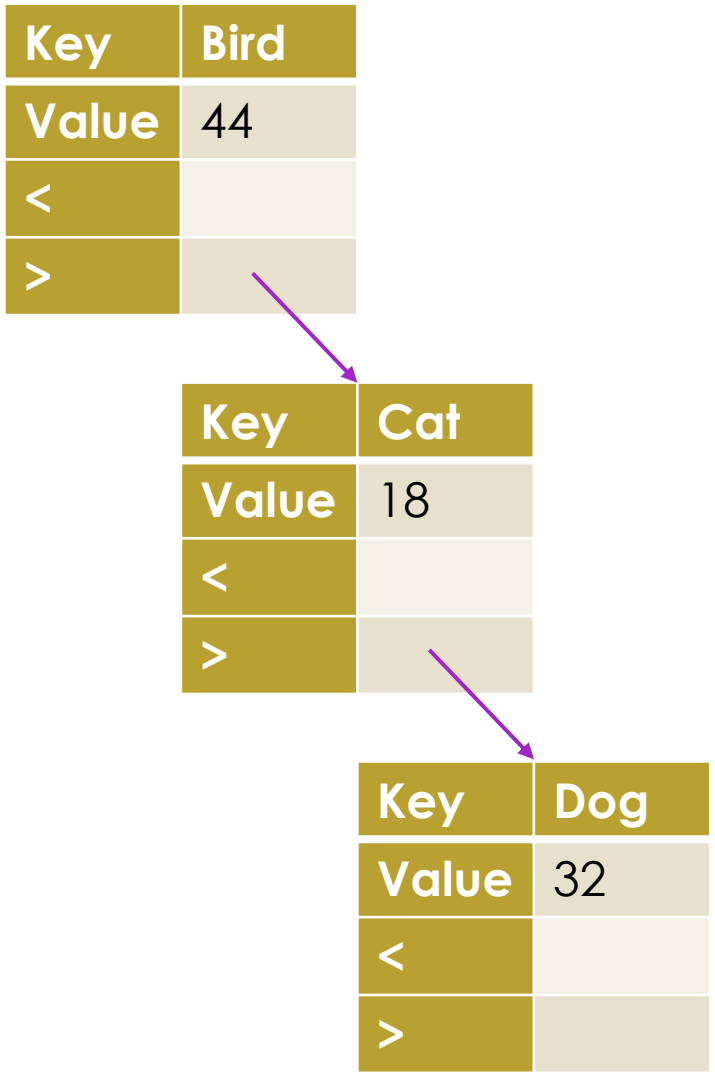
Binary Search Tree

- Binary Search built into list?
- Insert
  - O(n) worst case
  - Θ(log n) if balanced
- Search
  - O(n) worst case
  - Θ(log n) if balanced



Binary Search Tree

- Binary Search built into list?
- Insert
  - O(n) worst case
  - Θ(log n) if balanced
- Search
  - O(n) worst case
  - Θ(log n) if balanced



Pair Programming Exercises

- Work on the exercises in pairs
- Workshops this week and last week should be particularly useful for the assignment