



# CSSE 230 Day 30

That's all folks.

# Course Evaluations on Banner

- ▶ Numbers are nice, but written explanations are much better
- ▶ Focus:
  - Did you learn a lot?
  - Are there things you know/can do now that you didn't/couldn't at the beginning of the term?
  - What about the course/instructor enhanced your learning?
  - What about the course/instructor were barriers to your learning?
  - Be as specific as possible.

# Let's revisit the "Big picture", with understanding

## THE BIG PICTURE CSSE 230 – DATA STRUCTURES AND ALGORITHM ANALYSIS

Applications:  
(120/220 = 50% of 230 + beyond)

ADT:  
(120/220 = 50%)

List Stack Queue  
PQ

Set / Map (key/value)

Implementation  
Choices

(heap) (hash) (circular) (skip)  
Array Linked List  
(120/220 = 20%) (20 = 10%)

Tree  
(45%)

Graph  
(5%)

Diagram

Why use?  
(120/220)

Access by index:  
Search:  
Insert/remove  
from start/middle

Access by index:  
Search:  
Insert/remove  
at ends or from  
iterator:

Access by index:  
Search: \*  
Insert/remove: \*

(CSSE473)

Other notes:

Can sort!  
If sorted, search:

\*  $O(\log n)$  if balanced. Otherwise  $O(n)$

# Some Final Thoughts

- ▶ Data is at the heart of software.
  - The companies you may work for agree!
  - The data is the “irreducible complexity” of the code.
- ▶ This class has been very “heads down.”
  - Getting the algorithms right.
  - Making good OO design choices.
- ▶ You also need to be “heads up.”
  - Like the ethics assignment you did.
  - Understanding requirements means knowing the clients and users!
    - More on that in CSSE 371 ...

# Final Exam Details

- ▶ Thursday morning.
- ▶ Format same as previous exams.
- ▶ You can bring two sides of 8.5" x 11" paper.
- ▶ Comprehensive.
- ▶ Best preparation: Written problems

# Final Exam topics

- Reading, programs, in-class, written assignments.
- Foci:
  - Binary trees, including EBT, AVL, red/black, rank, and threaded trees
    - Traversals and iterators, numeric properties
  - PriorityQueues, Heaps and heapsort
  - Issues in Hash table implementation
  - Graphs
  - Recurrence relations
  - Sorting algorithms and analysis
    - Algorithm analysis ( $O$ ,  $\theta$ ,  $\omega$ ) in general
  - OO programming, using various data structures (lists, stacks, queues, sets, maps, priority queues)
    - +/– with ADT implementation options (like we did for PQ last week – be specific with answers)

# What's left?

- ▶ Finish sorting races by 11:59 PM Friday (late day until Saturday is OK)
  - SortingRaces eval is OPTIONAL
- ▶ Study, including taking the practice exam
  - ▶ Extra help meetings by appointment
- ▶ Final Exam Thursday