

Divide and Conquer

Dr. Mattox Beckman

UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN
DEPARTMENT OF COMPUTER SCIENCE

Objectives

Your Objectives:

- ▶ Know how to ensure quicksort doesn't go wrong.
- ▶ How to use the binary search principle on things other than searching arrays.

Two common forms

- ▶ Combining subproblems: break the problem space into parts, solve the parts, combine the parts. Example: sorting, segment trees
- ▶ Pruning search space: evaluate current situation, prune half of search space, search the other half. Example: binary search

Sorting Considerations

0 2 12 40 40...40 40 30 14 9
An array with lots of 40's

- ▶ For quicksort: you already know to pick a random pivot.
- ▶ You also need to partition into 3 spaces: $<$, $=$, $>$.
- ▶ Really, just use `sort` from the `algorithm`...
- ▶ Unless you need stable sorting.

Binary Search

```
0 #define EPS 1e-9 // Code from Competitive Programming 3 text
1 bool can(double f) {
2     // Code to do whatever it is you are trying to do
3 }
4 int main() {
5     double lo = 0.0, hi = 10000.0, mid = 0.0, ans = 0.0;
6     while (fabs(hi - lo) > EPS) { // answer not found yet
7         mid = (lo + hi) / 2.0;
8         if (can(mid)) {
9             ans = mid; hi = mid;
10        } else lo = mid;
11    }
12    printf("%.3lf\n", ans);
13 }
```