week4 section

Ezra Zigmond
ezigmond@college.harvard.edu

September 27, 2015

pset 1 feedback

- Be sure to run check50 and style50
- Try to use descriptive variable names (strlen $>$ 1 character)
- Use braces only when necessary (loops, functions, conditions)
- Comments

week3 Topics

- Asymptotic Notation (O and Ω)
- Binary search and its complexity
- Bubble, selection, insertion, and merge sort
- Understanding distribution code
- pset3

Asymptotic Notation

- We often want to formalize how fast a program runs or how much space it takes up in terms of the size of the input (n).
- We only really consider the case where n is very large
- big-O is an "upper bound"
- Ω is a "lower bound"
- This is different than "worst case" and "best case"
- You don't have to be able to derive the asymptotic complexity of algorithms, but you may have to explain informally.
- (If you want to learn more, take CS124)

Binary Search

- Idea: if your array is sorted, you can search in sublinear time
- Divide and conquer (phonebook example)
- What's the most times you'll have to divide?

Bubble Sort

- Probably the simplest $O(n^2)$ sort to implement
- What is the "worst case"?
- What is the "best case"?
- Interesting variation: comb sort (has the same worst case)

Selection Sort

- A very simple idea: find the smallest element and put it at the beginning

Insertion Sort

- Most similar to the way a human would probably sort, but slightly more difficult to implement in practice
- Given a choice (e.g. pset3) I would choose one of the others
- Make sure you understand the difference between insertion and selection

Merge Sort

- Idea: We can sort faster than $O(n^2)$
- Another divide and conquer approach
- Recursion
- You will probably never have to implement merge sort

Debugging

- It's a lot easier this year!
- A lot of the time, printf statements are good enough, but using the debugger is a lot more powerful
- Example