Searching

11-11-2011

Opening Discussion

- Minute essay comments:
  - You kids use computer search for EVERYTHING!
  - (Even answering how many seconds are in a year.)

Binary Search

- If the data is sorted, we can do something much better.
- We check the middle to see if it matches. If it does, return it. Otherwise, see if what we want is above or below the middle and repeat the process on only that half.
- This continually divides the things we are searching in half.
- Order?

Performance of Binary Search

- Dividing something by a fixed fraction repeatedly leads to O(log n) speed.
- O(log n) is much better than O(n) when n is large. To see this, consider a base 2 log for 1000, 1000000, or 1000000000.

Computer Memory

- The memory that a program uses is broken into two different parts.
  - Stack – This holds local variables. Every function/method call gets a new “frame” on the stack. Efficient, but limited.
  - Heap – All objects in Scala are allocated on the heap. It is big and flexible, but disorganized.
- Other languages allow you more direct control over memory. This has the potential to lead to errors.

Classification of Bugs

- We classify the errors that occur in programs in three broad groups.
  - Compile Errors – Found by the compiler. Gets a reasonable error message and line number.
  - Runtime Errors – Program crashes while running for a particular input. Gives type of error and line.
  - Logic Errors – Code runs fine, but does wrong thing. No information given to help you.
- You want to have your errors be higher up on this list because it gives you more information and makes it easier to fix.

Motivation

- We have been using “flat” text files to store things.
- Advantage: it is human readable and simple.
- Disadvantages: everything else.
    - Slow
    - Large
    - Lacks meaning
    - Hard to edit
    - Hard to debug

XML

- The eXtensible Markup Language (XML) is a standard for text encoding of data.
- If you have ever done HTML, XML is similar. XHTML is HTML that follows the XML standard.
- The advantage of XML is that it can encode pretty much anything and it is human readable text.
- The downside is that it can be very verbose.
- Composed of markup (between < and > or & and ;) or content (anything not markup).

Tags

- The primary markup used in XML is the tag.
- A tag begins with a < and ends with a >.
- There are three types of tags.
  - Start-tag: <student>
  - End-tag: </student>
  - Empty-element tag: <quiz/>

Elements

- The structure of XML documents comes primarily from elements.
- An element is one of the following:
  - Everything from a start-tag to the matching end-tag.
  - An empty-element tag.
- Elements have to be properly nested. The nesting can imply information.

Attributes

- An attribute is a name value pair.
- They can be put in start-tags or empty-element tags.
- Examples:
  - <student name=”Jason” id=”0123456”>
  - <quiz grade=”55”/>

XML Declaration

- An XML file can begin with a declaration telling information about it.
  - <?xml version="1.0" encoding="UTF-8" ?>
- We won't worry about these in this class.

Minute Essay

- What are examples when you use a computer to “search” in your life.