Today’s Outline

- Administrative Cruft
- Overview of the Course
- Some C++ topics

Course Information

- Instructor: Martin Dickey
  Office hours TBD
- TA Gerome Miklau
  Office hours: TBA
  - Please read before coming to class
- Probably there will be no course packet!

What is the Course About?

- Data structures – ways of organizing data in a program
  - Different applications call for different structures
- Each data structure is associated with a set of algorithms
- Each algorithm has properties (like speed) that can be investigated
- Many data structures are “classical”
  - Every programmer needs to know them. Why??
- Eventually you may need to invent your own

Observation

- All programs manipulate data
  - programs process, store, display, gather
  - data can be information, numbers, images, sound
- Each program must decide how to store data
- Choice influences program at every level
  - execution speed
  - memory requirements
  - maintenance (debugging, extending, etc.)
Goals of the Course

• Become familiar with some of the fundamental data structures in computer science
• Improve ability to solve problems abstractly
  – data structures are the building blocks
• Improve ability to analyze your algorithms
  – prove correctness
  – gauge (and improve) time complexity
• Become modestly skilled with the UNIX operating system and X-windows (you’ll need this in upcoming courses)

Course Activities

• Weekly written homework
• Projects (probably 4 total)
• 3 tests (maybe only 2??), including the final exam
  – Final is August 18 – no exceptions!
• In-section quizzes
  – Occasional, cover recent material, similar to homework exercises
• In-lecture quizzes
  – Frequent, maybe even daily
  – cover assigned reading for that day!

Course Mechanics

• 326 Web page: www/education/courses/326
• 326 course directory: cse/courses/cse326
• 326 mailing list: cse326@cs
  – subscribe to the mailing list using majordomo, see homepage
• Course laboratory is 329 Sieg Hall
  – lab has NT machines w/X servers to access Linux
    (UNIX)
• All programming projects graded on Linux/g++

Some Review Topics

• Are you ready?? See the “preconditions”
• C++ review topics
  – If you studied it in 143, you will need it in 326
  – Classes are the big item
  – Review textbook section 1.5 carefully
  – May be some new topics buried there
• CSE 321 topics
  – mathematical induction
  – series
  – counting and probability

New C++ Topics

• Some new topics
  – Templates
    • some messy notation, but a powerful idea
    • templated functions
    • templated classes
  – The Standard Template Library (STL)
    • “container” classes
    • iterators
    • generic (templated algorithms) algorithms

vector: An Array on Steroids

To use:
#include <vector>
Then:
vector<double> fiblist (100);
vector<StudentRecord> students (1000);
vector<char *> monthnames;
vector<string> daynames(7);
Then:
etc.
Reading for Next Time

- “Catch up” by reading review parts of chapter 1 (through 1.5)
- Templates are covered in section 1.6.
- Prepare for next two lectures by reading into chapter 2 as far as you can.
- If there was a quiz Weds, it would cover material from 1.6 and 2.1