Final Exam Review

CS 2308
Fall 2016

Jill Seaman

Final Exam

- section ~~001~~ 002 (12:30pm): Wed 12/14 11am-1:30pm
- section ~~002~~ 001 (2:00pm): Wed 12/14 2pm-4:30pm
- In your regular classroom
- Closed book, closed notes, clean desk
- Comprehensive (covers entire course)
- 30% of your final grade
- Bring your ID card!
- Bring a number 2 pencil and eraser.
- No calculators, no headphones/earphones

Exam Format

- 150 points total:
  - 74 pts:
    - Multiple choice (scantron form)
    - 37 questions
  - 76 pts:
    - Writing programs/functions/classes/code
    - Finding errors in code (2 pages)

Content

150 points total (approximate break down):

- Unit 1 (23pts)
- Unit 2 (13pts)
- Analysis (12pts)
- Unit 3 (20pts)
- Unit 4 (20pts)
- C++ Programming on Linux (12pts)
- Unit 5 (20pts)
- Unit 6 (30pts)

Unit 1: Functions, Arrays & Structs

- Know how to program with functions, arrays and structures.
- Passing parameters by reference (and value)
- Scope rules
- Overloaded functions and default arguments
- Be able to process arrays
  - Be able to find the minimum/maximum value!
  - See review exercises
- Be able to find errors in code

Unit 2: Searching, Sorting & Analysis

- Searching
  - Linear Search
  - Binary Search
- Sorting
  - Bubble Sort
  - Selection Sort
- See review exercises:
  - Sample exercises to demonstrate algorithms
  - Be able to code linear search

Analysis of Algorithms: efficiency

- Efficiency
  - Growth rate functions, which are faster/slower
  - Use big-O notation
  - Efficiency of
    - searching/sorting algorithms
    - array access and traversal
    - linked list operations
  - See the Final Exam Review Exercises for good coverage on this

Unit 3: Pointers & Dynamic Memory Alloc

- Pointer variables: how to define + initialize
- Address of (&) and Dereferencing (*) operators
- Pointers and arrays
  - an array variable is the address of its first element
  - array[index] = *(array + index)
- Dynamic memory allocation
  - new + delete operators
  - allocate new arrays (duplicateArray, etc.)
- Pointers as parameters (call by reference, arrays)
- Using pointers with linked lists

Unit 4: Intro to Classes

- Procedural programming vs object oriented programming
- Encapsulation, Data hiding, Interface
- Fundamentals of classes and objects:
  - Members: variables and functions
  - private vs public, access rules
  - declaration and implementation of classes
    - class declaration
    - defining member functions
  - constructors and destructors
  - instances and the dot operator

Unit 4: Intro to Classes

- Fundamentals of classes and objects (cont.):
  - inline member function definitions
  - defining instances of a class (objects)
  - arrays of objects, initialization
- Pointers to objects (and structs)
  - how to declare, assign
  - using ->
  - p->member vs. (*p).member vs. *(p.member)
  - dynamic allocation of objects, structures
  - when destructor function is called

C++ Programming on Linux

- Basic shell commands, know how to use
- edit, compile, run (nano, g++, a.out)
- Compiling multiple files:
  - How to split up code, what goes where
  - g++ a.cpp b.cpp
  - separate compilation
    - g++ -c a.cpp
    - g++ -c b.cpp
    - g++ a.o b.o
  - makefile: understand the ones used for the assignments, know how to use them

Unit 5: Linked Lists

- Dynamically allocated list data structure
- Organization: nodes, head pointer, empty list, NULL
- Linked list tasks: T1-T11:
  - create empty list, create a new node
  - add to front of list
  - append to end of non-empty list
  - traversing a linked list (display, count, sum, etc)
  - how to advance 2 pointers together (n and p)
  - delete given n and p, special cases
  - insert given n and p, special cases
  - linked list destruction
- Arrays vs Linked Lists

Unit 6: Stacks and Queues

- Know what ADT, LIFO and FIFO mean
- Know the 4 basic operations of each data type:

pop	enqueue	isEmpty
push	dequeue	isFull

- Be able to show contents of stack or queue after a series of operations
- Be able to implement the operations using a static array or a linked list.
- Be able to use a stack to solve a problem like matching brackets.

Sample Problems

See the lecture notes titled:

Final Exam Exercises

on the website

Office Hours
finals week

Day	Date	Time
M	12/12	1:00-2:30pm
T	12/13	2:30-4:00pm
W	12/14	none (exams)
Th	12/15	none
F	12/16	2:30-3:30pm
		and by appt!

How to Study

- Start with the topics from this set of slides (Final Exam Review).
- Use the regular semester lectures to make sure you understand the topics (quiz yourself).
- Use the textbook to make sure you understand the lectures about the topics.
- **Do** the review exercises on the Final Exam Exercises slides. Do book exercises. Practice!!
- Go over the exams and assignment solutions (fix yours).
- Discuss with others! (and get some sleep)