Final Exam Review

CS 2308
Fall 2013

Jill Seaman

Final Exam

- Section 001(MW):Mon, Dec 9, 2:00PM to 4:30PM
- Section 003(TR):Tues, Dec 10, 11:00AM to 1:30PM

- Closed book, closed notes, clean desk
- Comprehensive (covers entire course)
- 30% of your final grade
- I recommend using a pencil (and eraser)
- I will bring scratch paper.
- No calculators.

Exam Format

- 200 points total (10 or 11 pages):

  * Writing programs/functions/classes/code
  * Multiple choice
  * Tracing code
  * Tracing search/sort algorithms
  * Finding errors in code
  * Short answer (some very short, some longer)

Example Problems

See the lecture notes titled:

Final Exam Exercises

on the website

Chapters 1-7 Review

- Know how to program with arrays and functions
- Passing parameters by reference (and value)
- Passing arrays to functions
- Overloaded functions
- 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

5

Ch.8: Searching and Sorting Arrays

- Searching
    - Linear Search
    - Binary Search
- Sorting
    - Bubble Sort
    - Selection Sort
- See review exercises:
    - Describe algorithms in English
    - Sample exercises to demonstrate algorithms

6

Analysis of Algorithms: efficiency

- Efficiency
    - Growth rate functions, which are faster/slower
    - Use big-O notation
    - Efficiency of
        ‣ searching/sorting
        ‣ array access and traversal
        ‣ linked list operations

    - See the Final Exam Review Exercises for good coverage on this

7

Ch 9: Pointers

- 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
    * allocate new arrays (Programming Assignment 3)
- Pointers as parameters (call by reference, arrays)
- Using pointers with linked lists

8

Ch. 10: Strings and Things

- Char functions: isalpha, isdigit, islower, tolower, ...
- C-strings:
  - representation:
    - char array
    - terminated by null character ('\0')
  - library functions: strlen, strcpy, strcmp
- Predefined string class:
  - initializing string objects.
  - operations: =, <<, >>, relational ops, [n]
  - member functions: length, size, append, at
- Be able to write functions that process string data.

9

Ch 11: Structures

- Structures:
  - Definition (new data type) and variables
  - How to access members (dot operator)
  - Operations (which are valid)
  - Arrays of structures
  - Pointers to structures (-> operator):
    - s->x   (*s).x   *(s.x)
  - Dynamic memory allocation
  - Use of structures in linked lists (Nodes)

10

Ch.13+14: 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
    - defining member functions
    - overloaded operators
  - constructors and destructors
  - copy constructor (default)

11

Ch.13+14: Classes

- Fundamentals of classes and objects (cont.):
  - inline member function definitions
  - instance variables vs static variables
  - defining instances of a class (objects)
  - arrays of objects, initialization
- Pointers to objects
  - how to declare, assign
  - using ->
  - dynamic allocation of objects
  - when destructor function is called
  - the "this"pointer

12

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

13

Ch. 17: Linked Lists

- How to define a linked list (node declaration and head pointer definition).
- Adding a node (insert at front or append)
- Insert or delete node from the middle of a list
  - how to advance 2 pointers together
  - be able to set pointers in general case
- How to traverse a linked list to
  - display it, calculate some value
  - find minimum/maximum
  - find last node
- Arrays vs Linked Lists

14

Ch. 18: Stacks and Queues

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

pop	enqueue
push	dequeue
isEmpty	isEmpty
isFull	isFull

- Understand how to use a stack to perform algorithms done in class
- Be able to show contents of stack or queue after a series of operations
  (see Final Exam Review Exercises)

15

Office Hours
after last class day

Day	Date	Time
M	12/9	12:30-1:30pm
T	12/10	9:30-10:30am
W	12/11	11:00am-12:00noon
Th	12/12	1:30-3:30pm
		and by appt.

16

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.

- Use the textbook to make sure you understand the lectures about the topics.

- Do the review exercises on the Final Exam Exercises slides.

- Go over the midterm exams and assignment solutions.

- Discuss with others!

17