Exam 1 Review

CS 2308
Fall 2016

Jill Seaman

Exam 1

- Monday, October 17
- In class, closed book, closed notes, clean desk
- 20% of your final grade
- 80 minutes to complete it
- Bring your ID card!
- Bring a number 2 pencil and eraser.
- NO: calculators or cell phones.
- NO: headphones/earbuds.

Exam Format

- 100 points total
  - 50 pts: 25 Multiple choice/matching (scantron form)
  - 18 pts: Demonstrating the search/sort algorithms
  - 32 pts: Writing functions/code (about 2 pages)

Content from Textbook

- Chapter 6: 6.1-10, and 13
- Chapter 7: 7.1-4, 6, and 8
- Chapter 11: 11.2-8
- Chapter 8: 8.1 and 8.3
- Chapter 9: 9.1-9
- Linux material from the Linux lecture (slides 1-13)
- see Unit outlines and lectures for specific topics: Unit 1, Linux, Unit 2, Unit 3

Unit 1: Functions, Arrays & Structs

- Passing parameters by reference and by value
- Scope rules
- Passing arrays to functions, processing arrays
- Partially filled arrays
- Arrays of structures
- Overloaded functions and default arguments (definitions only)
- Be able to write code with functions, arrays and structures.   (Be familiar with PA1 and PA2).

C++ Linux

- What is Linux?
- Linux file system
- Basic shell commands

```
pwd      more/less/cat
ls       cp
cd       mv
mkdir    rm
rmdir    man
```

- Basic file editing (nano, etc.)
- edit, compile, run

```
nano
g++
a.out
```

- know how to **use** the commands

Unit 2: Searching, Sorting & Analysis

- Searching
  - Linear Search
  - Binary Search
- Sorting
  - Bubble Sort
  - Selection Sort
- Efficiency
  - Growth rate functions: which are faster/slower
  - Efficiency of each searching/sorting algorithm

You **will not** need to know the code
--but you need to be able to implement linear search

You **will** need to be able to demonstrate the algorithms
--see exercises at end

Unit 3: Pointers & Dynamic Memory Allocation

- Address operator (&)
- Pointer variables: how to define (data type)
- Dereferencing operator (*)
- Pointers and arrays
  - an array variable is the address of its first element
  - array[index] = *(array + index)
- Pointer arithmetic (if ptr points to a var of type d):
  - ptr + n = address in ptr + n * sizeof(d)
- Initializing Pointers

Unit 3: Pointers & Dynamic Memory Allocation, cont.

- Comparing pointers
- Pointers as function parameters
  - Pass by reference using pointers as parameters
  - Pointers used as parameters accepting arrays as arguments
- Dynamic memory allocation
  - new operator
  - new with arrays
  - delete
  - return pointers from functions (duplicateArray)

Be able to write code to solve problems like the ones on PA#3.

Sample Problem: multiple choice

Given the following program:

```
int main () {
   int *ptr1;
   int foo1 = 42;
   int x[ ] = {10,20,30};

   ptr1 = &foo1;
   *ptr1 = 13;

   cout << "A- " << foo1 << endl;
   cout << "B- " << ptr1 << endl;
   cout << “C- “ << *(x+1) <<endl;
}
```

If the output includes an address, choose (e) something else

1. **(2 pts)** What is the output of this program on the line labeled A?

   (a)A- 42 (b)A- 13 (c)A- 10   (d)A- 20 (e) something else

2. **(2 pts)** What is the output of this program on the line labeled B?

   (a)B- 42 (b)B- 13 (c)B- 10   (d)B- 20 (e) something else

3. **(2 pts)** What is the output of this program on the line labeled C?

   (a)C- 10 (b)C- 11 (c)C- 20 (d)C- 30 (e) something else

Demonstrating Searching
Example

The target of your search is 101. Use the following array of integers (values on top, indexes on bottom):

1	7	8	14	20	42	55	67	78	101	112	122	170	179	190
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

What are the first three values (in order) that the target value 101 will be compared to during

A. Linear Search?
B. Binary Search?

What are the first four values (in order) that the target value 114 will be compared to during

A. Linear Search?
B. Binary Search?

Demonstrating Sorting
Example

Use the following array for both questions (values on top, indexes on bottom):

11	8	14	7	12	18	2	17
0	1	2	3	4	5	6	7

Show the contents of the array after 2 passes of the selection sort

Show the contents of the array after 1 pass of the bubble sort

Example Programming Problem

Given the following struct definition:

```
struct Player {
   string name;
   int number;
   int points;
};
```

Write a function called addPlayer that takes 2 arguments: a partially filled array of Player and the number of elements it currently contains (count). It should add one player to the array by inputting the necessary values from the user (assume the name has no spaces).

How to Study

- Review the slides and Top Hat questions
  - understand all the concepts, quiz yourself
- Use the book to help understand the slides
  - there will be no questions over material (or code) that is in the book but not on the slides
- Review programming assignments (fix yours!)
  - get printouts of solutions in my office
- Do the practice exercises in Unit outlines!!
- Practice, practice, practice!
- Get some sleep