

Exam 1 Review

CS 2308
Fall 2011

Jill Seaman

1

Exam 1

- Thursday, October 6
- In class, closed book, closed notes, clean desk
- 20% of your final grade
- 80 minutes to complete it
- I recommend using a pencil (and eraser)
- I will bring scratch paper.
- No calculators.

2

Exam Format

- 100 points total
 - * Plenty of writing programs/functions/code
 - * Some combination of:
 - Multiple choice
 - Fill-in-the-blank
 - Tracing code/finding errors in code
 - Short answer

3

Example Programming Problem

Write a function that accepts an array of integers and the size of the array and prints out a table listing how many values in the array fall in each of the following ranges:

less than 50

50 to 59

60 to 69

70 to 84

85 to 99

over 100

4

Example Tracing Problem

What will the EXACT output of the following program be?

```
#include <iostream>
using namespace std;

int main () {
    int *ptr1, *ptr2;
    int fool, foo2 = 13;

    fool = 42;
    ptr1 = &fool;
    ptr2 = ptr1;

    cout << "*ptr1 - " << *ptr1 << endl;
    cout << "fool - " << fool << endl;
    cout << "*ptr2 - " << *ptr2 << endl;

    *ptr1 = 100;
    *ptr2 = 200;

    cout << endl;
    cout << "*ptr2 - " << *ptr2 << endl;
    cout << "what? " << fool%10 << endl;

    return EXIT_SUCCESS;
}
```

5

Chapters 1-7 Review

- Know how to program with arrays and functions
- Passing parameters by reference
- Passing arrays to functions
- Understand Programming Assignment 1

6

Ch.8: Searching and Sorting Arrays

- Searching
 - Linear Search
 - Binary Search
- Sorting
 - Bubble Sort
 - Selection Sort
 - Quicksort (algorithm and efficiency only)
- Efficiency
 - Growth rate functions, which are faster/slower
 - Efficiency of searching/sorting

7

Ch 11: Structured Data

- Data Types:
 - Set of values + operations
 - Scalar vs composite data types
- Structures:
 - Definition (new data type)
 - Variable definitions
 - How to access members (fields)
 - Operations (which are valid)
 - Arrays of structures
 - Nested structures
 - Structures as function args, return values

8

Ch 9: Pointers

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

9

Ch 9: Pointers, 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 (watch outs) ¹⁰

C++ Programming on Linux

- What is Linux, Linux file system
- Basic shell commands
- edit, compile, run (nano, g++, a.out)
- Programming with multiple files
 - * Why **and** How to split program up
 - * Header files
 - * How to compile multiple file program:
 - g++ (g++ a.cpp b.cpp)
 - separate compilation (g++ -c a.cpp, g++ -c b.cpp, g++ a.o b.o)
 - makefile: understand the example

11

How to Study

- Start with the slides/presentations
- Read book to understand slides
- Review assignments + solutions
- Do some exercises from the book
 - * Fill-in-the-Blank
 - * Algorithm workbench
 - * Find the Error
 - * Programming Challenges (first few of these)
 - * T/F
 - * Short Answer

12