Exam 2 Review

CS 1428
Fall 2017
Jill Seaman

Exam 2

- Thursday, November 16
- In class, closed book, closed notes, clean desk
- 15% 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 points: 25 multiple choice and T/F (scantron form)
  ‣ 50 points: writing code on the test paper
    ➡ programs, functions and individual statements
- Tasks:
  ‣ Tracing code (what is the output)
  ‣ Finding errors in code
  ‣ Demonstrate general knowledge about C++ and programming
  ‣ Programming (writing code)

Content from Textbook

Units 4 through 6:
  - Chapter 5: 5.2-12
  - Chapter 6: 6.1-5, 7-10, 13
  - Chapter 7: 7.1-4, 6 and 8

Primarily loops, arrays, and functions
Loops

- while loop
  - general purpose
- do-while
  - body always done once
  - good for menus, repeating a process
- for loop
  - init; test; update
  - all are optional
- Which loops are good for which situations

Arrays

- Array declaration/definition:
  - int list[10];
  - size declarator must be a constant (in the C++ standard)
- Array elements
  - list[i]
  - range of subscripts
  - types
- Array initialization:
  - int list[] = {6,7,8};

Loops

- Using a while loop for input validation
- Counters/count controlled loop
- Keeping a running total
- Sentinel controlled loop
- Nested loops
- Reading data from a file of unknown length
  - while (fin >> number)
- Break and continue
- Infinite loops

Arrays

- Processing arrays
  - input and output
  - sum, average
  - finding max, min (and index of which one)
  - counting values that pass a test
  - array assignment (copy)
- Lack of bounds checking
Functions

- Function definition
  - name, return type, parameter list, body
- Function call
  - name, argument list
- Function prototype, when it is required
- Function parameters and arguments
  - Understand how they work

Functions

- The return statement
  - returning a value from a function
  - calling a function that returns a value
- Pass by value
- Pass by reference
- Scope and Lifetime
  - local and global variables
  - parameters
  - global constants

Functions and Arrays

- Passing array **elements** to functions
  - parameter type matches element type
- Passing **entire** arrays to functions
  - parameter type is an array (no size declarator)
  - separate int parameter for size (usually)
  - argument is name of the array (no brackets)
  - arrays are ALWAYS passed by reference

Software Development Process

- Top Down Design
  - Break tasks into subtasks
  - Make a hierarchy of tasks
- Incremental Development
  - Implement one piece at a time
- Testing
  - Test cases: input values and expected output
- Debugging
  - Strategy: output values of variables
  - Strategy: output literals to trace execution path
Sample problem: multiple choice

- What is output by the following statements?

```cpp
int list[] = {8,10,3,55,1,22,3,17};
int x = 10;
int i = 3;
while (i < 8) {
    int t = list[i];
    if (t < 10) {
        x = 7;
    } else if (t < 20) {
        x++;
    } else {
        x--;
    }
    i = i+3;
}
cout << "x = " << x << endl;
```

A) \( x = 10 \)
B) \( x = 7 \)
C) \( x = 8 \)
D) \( x = 6 \)

Sample problem: Programming

The formula for the volume of a sphere is

\[
A = \frac{4}{3}\pi r^3
\]

where \( \pi \) is 3.14159 and \( r \) is the radius of the sphere. Write a function named volume that accepts a radius as an argument. The function should return the volume of a sphere having that radius. Demonstrate the function by calling it in a loop in the main function that displays a table of volumes of circles with radius values 1 through 10. Your function should work properly if the radius has a fractional part (i.e. 5.89).

How to study

- Review the slides (these, and Units 4 - 6)
  - understand all the concepts, quiz yourself
- Use the book to help understand the slides
  - there will be no questions over material that is in the book but not on the slides
- Review programming assignments (fix yours!)
  - get printouts of solutions 4, 5 and 6 up front or in my office
- Review the Top Hat questions
- Try some exercises from the book
- Practice, practice, practice! Write code! Sleep!