



Oregon State
University

COLLEGE OF ENGINEERING

| School of Electrical Engineering
and Computer Science

CS 161

Introduction to CS I

Lecture 12

- What we learned from Midterm 1
- Variable scope
 - Pass function arguments by value or reference
- Assignment 3 tips



Midterm 1 Solutions

- **Even if you don't know some of these now... you will!**
 - Errors highlight for you what to focus on studying
- Average: 84%
- Solution (form K) is posted on course website
 - Calendar -> 02/03 (lecture 12)
 - Direct link:
http://classes.engr.oregonstate.edu/eecs/winter2020/cs161-020/calendar/CS161_Midterm_1_W20_form1_soln.pdf

Midterm 1

- Part I
 - 3: operator precedence
 - 6: random number ranges – what does `rand()` return?
- Part II
 - 1: integer math
 - 2: Linux command from Lab 2
 - 3: operator precedence
 - 4: data type min/max values

Operator precedence

++	--
!	
*	/ %
+	-
<	<= > >=
&&	
==	!=
=	

Midterm 1

- Part II (cont.)
 - 5: no curly braces – this is okay (but only uses first such line)
 - 6: no break statements
- Part III
 - 6: any option is good if you have a good reason for it
 - 7: short-circuit only happens if evaluating one expression renders the next one **irrelevant**
 - 8: operator precedence and what is %?
 - 9: variable scope

Midterm 1

- Part III (cont.)
 - 10: integer math; stop for loop when $x < 0$ is false
 - note: loop counter update happens **after** iteration completes
 - 11: check for even values
 - 12: start at 'r' and go backwards to 'n'
 - 13: stop for loop if $x \geq 10$ **or** if $x > 3$ (break)
 - 15: nested loop: a has the same value until b loop finishes
 - 19: you can loop on one variable (m) and output a different one (num)
 - What is the value of a loop counter when the loop ends? (condition is false)

Midterm 1: Extra credit question 1

- "without using a **conditional** statement" means no if-then or switch; for/while/do-while ok
- New extra credit item on Canvas: "Midterm 1 extra credit"
 - Due midnight Tuesday, Feb. 4

Midterm 1: Also...

- If-then **vs.** switch – you can use an expression with switch if your cases are true **and** false.
You will however get a compiler warning.

```
switch (10 > 8) {  
    case true: cout << "10 is > 8" << endl; break;  
    case false: cout << "10 is not > 8" << endl; break;  
}
```

- Conditional/ternary operator: these lines are equivalent
 - `k = (3 < 4) ? 10 : 20;`
 - `if (3 < 4) { k = 10; } else { k = 20; }`
- Semi-colon is required at end of **statement**, not **line**

Study sessions start this week

- Thursdays, 6-7 p.m., LINC 268
- Print out the worksheet (from course website calendar) and complete it in advance
- http://classes.engr.oregonstate.edu/eecs/winter2020/cs161-020/calendar/WS5_W20.pdf
- The TA study session leader will go over the answers

Assignment 3 tips

- **No global variables, no goto**
- C++ buffers output; if you print only a single character it may not show up for a while (until next `endl`)
 - Solution: use `flush` to "flush" the buffer immediately
 - This is actually `std::flush`
- How to modify the current line of output, instead of going to the next line?
 - Use `\b` (backspace) in your string
 - Use `\r` (carriage return) in your string (and `flush`)

Week 5 begins!

- Attend lab 5 (laptop required) – Practice Proficiency Demo
- Read Rao Lesson 7 (pp. 166-167) – functions
Read Rao Lesson 8 (pp. 205-210) – references
- (Optional) Attend Study Session
(Thursday, 6-7 p.m. in LINC 268)
- Continue working on **Assignment 3 implementation**
(due Sunday, Feb. 9)

See you Wednesday!