1 Introduction

This lab lets you practice writing boolean expressions and if statements. Sit down, log in, and start up IDLE on your favorite computer.

The first set of tasks uses the interpreter.

2 Boolean Expressions

Try typing these statements to check your understanding of relational and boolean operators. Be sure you can evaluate it “by hand” to understand why each expression has the value it does.

```python
>>> x = 33
>>> y = 0
>>> z = False
>>> x <= y or not z
>>> not ( x > y and z)
>>> x == x # this is called a tautology: it doesn’t matter what x is, it’s always true
>>> x and y # mixed types: note the result
>>> y and x # mixed types: note the result
```

Try typing your own boolean expressions to solve the following problems.

1. Write a boolean expression that is True if \( x \) is positive (greater than zero), and \( y \) is an even number; otherwise the expression is False.\(^1\) Note: a number is even if the remainder when dividing by 2 is equal to zero. Does that remind you of anything you learned last week?

\(^1\)I’m going to quit saying this. From now on, assume that any instruction about what the boolean expression does ends with “otherwise the expression is false.”
Test your boolean expression with four combinations of values for x and y, to create four possibilities for the and: (False, False) (False, True) (True, False) and (True, True). Does it work?

2. Now write an if statement that uses your boolean expression. If the expression is True it prints: Wow!. If the expression is False it does nothing.²

Note the interpreter automatically indents the code block after the if statement: you have to type a blank line to indicate the end of the block. The if statement won't execute until you've typed the whole block and the blank line.

3. Now write the if statement again, but add an else part so that if the expression is False, it prints Invalid inputs!.

Note you have to out-dent before typing else, using the Delete or Backspace key. The interpreter will automatically indent after the else, and you type a blank line to indicate you are finished with the block.

4. Now add an elif to the above statement. Note the syntax rule: an elif part has to go between the if part and the else parts. Your three-way if statement should work like this: If both x is positive and y is even, it prints Wow!; otherwise, if one of these things is true but the other is false, it prints Almost!; otherwise (both are false), it prints Invalid inputs!.

5. Now make an if statement (possibly a combination of if statements) that works like this: if x is greater than or equal to y but no more than twice y, it prints x is big but not too big; if y is greater than than x but no more than twice x it prints y is big but not too big; in all other cases it prints nothing at all.

So, depending on the values of x and y, the user could see both print messages (when x == y); or just one of the print messages (when x ≥ y or y ≥ x), or neither of the print messages (if the bigger one is more than twice the smaller one). Try your statement with different values to make sure you got it right.

3 The Lab Assignment

- Point your browser to ...
  https://app.cs.amherst.edu/sfkaplan/courses/2014/spring/COSC-111/projects/lab3.py

²From now on, assume that if I don’t say anything about what to do if the expression is False, the if statement is not supposed to do anything.
...and save a copy on your U drive. Open it inside IDLE and run it to see what it does (not much, at this point).

- There are three parts to this assignment, as described in comments in the code. Be sure to get each individual part running before you go on to the next one.

- If you don’t finish before lab time is over, you can finish it ...later and go to ...
  https://www.cs.amherst.edu/Submit
  to turn it in. Your program is due by 9:00 Monday morning.