

CS 1110, LAB 11: WHILE-LOOPS

<http://www.cs.cornell.edu/courses/cs1110/2014fa/labs/lab11.pdf>

First Name: _____ Last Name: _____ NetID: _____

This lab is designed to give you practice with using while-loops. It includes both for loops over ranges integers and over more complicated conditions. With both the exam and the assignment due in the next week, we have tried to keep the demands of this lab reasonable.

Lab Materials. Download the appropriate files from the Labs section of the course web page.

<http://www.cs.cornell.edu/courses/cs1110/2014fa/labs>

When you are done, you should have the following two files.

- `lab11.py` (the primary module for the lab)
- `test_lab11.py` (a unit test for `lab11.py`)

Create a *new* directory on your hard drive and download all of the files into that directory. You can also get the files bundled in a single ZIP file called `lab11.zip` from the course web page.

Getting Credit for the Lab. The lab includes both written exercises and coding exercises. The latter requires that you modify the file `lab11.py`. We have provided a unit test, but you do not need to modify this file. When you are done, show both this handout and the file `lab11.py` to your instructor. Your instructor will then swipe your ID card to record your success.

As with all previous labs, if you do not finish during class time, you have **until the beginning of lab next week to finish it**. You should always do your best to finish during lab hours. Remember that labs are graded on effort, not correctness.

WRITTEN EXERCISES

Ranges of Integers. How many values are in the following ranges? The last few requires a formula in terms of h and k . Remember that in the notation $h..k$, we require $k \geq h-1$. For example, $5..4$ is OK but $5..3$ is not allowed.

Given Range	Contents	Given Range	Contents
$5..7$		$h..h+1$	
$5..6$		$h..h$	
$5..5$		$h..h-1$	
$5..4$		$h..k$	
$4..4$		$h-1..h+1$	

Assigning to Range Variables. Each line below asks you to write an assignment. We have done the first one for you to give you an idea of what we are looking for.

Range	Want	Assignment Statement
h..k	Assign to k so that the range has 1 element	k = h
h..k	Assign to h so that the range has 1 element	
h..k	Assign to k so that the range has 0 elements	
h..k	Assign to h so that the range has 0 elements	
0..n-1	Assign to n so that the range has 1 element	
0..n-1	Assign to n so that the range has 0 elements	
h-1..10	Assign to h so that the range has 1 element	
h+1..10	Assign to h to that the range has 0 elements	

CODING WHILE-LOOPS

Inside of `lab11.py` you will see the specifications of four functions. Implement **the first three of the functions in this module**. The **last function `exp` is optional**. Each implementation must contain a while-loop. You will notice that the first two functions are from Lab 8, the recursion lab. This is not a mistake; we want you to implement them differently using recursion. The third function is a variation on the second, and is easier to implement with a while-loop than recursion.

The optional function (which you do not have to implement) is the trickiest, as assumes some knowledge of Calculus. It is an approximation of e^x using Power Series. The Power Series for e^x is

$$e^x = \sum_{n=0}^{\infty} \frac{x^n}{n!} = 1 + x + \frac{x}{2} + \dots + \frac{x_n}{n!} + \dots$$

The values $\frac{x_n}{n!}$ are the *terms* of this Power Series. To approximate this sum, we just stop:

$$e^x \approx \sum_{k=0}^n \frac{x^k}{k!} = 1 + x + \frac{x}{2} + \dots + \frac{x_n}{n!}$$

The error of this approximation is the absolute value of the next term, $|(x^{n+1})/(n+1)!|$. You should write a while-loop that computes each term and adds it to the accumulator for e^x . It stops adding terms when they are greater than the error value.

For all of these functions, we have provided a unit test. You do not need to add any new tests. Simply use it to check you answers. When you are convinced that everything is working correctly, show your instructor what you have.