COMP220 — Lab 1 & Homework 1

Fall 2017

Abstract

For lab you'll review some basics of complexity and algorithm analysis as well as hone your skills with Logarithmic thinking. For homework you'll dust off your C++ skills and explore the new programming environment: *Code::Blocks*.

1 Lab 1

1. The formula for the conversion from one logarithm base to another tells us that for any two bases a and b, $\log_a(n)$ differs from $\log_b(n)$ by exactly a factor of $\frac{1}{\log_b(a)}$.

$$\log_b(n) = \frac{\log_a(n)}{\log_a(b)}$$

We're often interested in logs base 2, so let's compute the conversion factor for base $b = 2$ for each bases listed in the table below:

Base a	Factor to convert to $\log_2$
e	
8	
10	
16	
23	
127	
256	

What do you notice about converting to $\log_2$ from some base $b = 2^k$?

2. Fill in the following table in three phases. First without using a calculator, fill in the integer lower and upper bound. If the logarithm is an exact integer, then put that as both the lower and upper bound. Then, take a shot a guessing the logarithm's actual value out to 2 decimal places. Finally, use a calculator to compute the actual log to 2 places and fill that in for the final column. When computing actual logarithms, you may only use the log base 10 function on a calculator; use the conversion formula for all other bases.

Logarithm	Lower Int.	Upper Int.	Guesstimate	Actual Value
$\log_2(17)$				
$\log_2(725)$				
$\log_2(1024)$				
$\log_2(7)$				
$\log_8(7)$				
$\log_8(178)$				
$\log_{16}(25)$				
$\log_{16}(333)$				
$\log_{10}(15)$				
$\log_{10}(150)$				
$\log_{10}(145787)$				

3. Rank the following complexity from least to greatest order, in terms of resources needs by members of the class, where 1 is the least resource intense class and numbers proceed up from there.

$O(n \log n)$ _____

$O(n^2)$ _____

$O(1)$ _____

$O(2^n)$ _____

$O(n^3)$ _____

$O(\log n)$ _____

$O(n)$ _____

4. Determine the Big-O for each of the following functions.

$\frac{7}{100}n + 1.5E16$ _____

$\frac{x \log x}{25} - 13x + 5x^3$ _____

$\frac{(n+1)(n+2)}{n}$ _____

10^9 _____

$\log y + 135 + \frac{y}{2}$ _____

5. Consider the following loop template containing integer constants a and s:

```
for(int i{a} ; i < n ; i+=s){
}
```

(a) What constraints must be placed on the values of a, s, and n in order for this loop to terminate properly?

(b) Assuming the above constraints are met, exactly how many iterations will it perform before it terminates?

(c) What is the complexity of this loop and what effect, if any, do the values of a and s have on its complexity?

6. Consider the following loop template containing integer constants a and s:

```
for(int i{a} ; i > n ; i/=s){
}
```

(a) What constraints must be placed on the values of a, s, and n in order for this loop to terminate properly?

(b) Assuming the above constraints are met, exactly how many iterations will it perform before it terminates?

(c) What is the complexity of this loop and what effect, if any, do the values of a and s have on its complexity?

2 Homework 1

The goal of this homework is to get back into C++ and to familiarize yourself with the use of the new IDE, Code::Blocks. To do this you should complete *exercise 7 from chapter 2 of the text* with a few modifications:

1. Be sure to put your function in a library.
2. Write tests for your function using the gTest testing framework.
3. Write a main program that provides a basic CLI interface to your function. If your program is called *sqrt* then running *sqrt 4* at the CLI should produce *2*. You may assume valid command line arguments for this assignment; Error checking and validation of command line arguments is not required but you are strongly encouraged to do it anyway.

Submit your *source code only* via the *handin* program. The course is, of course, *comp220* and the assignment is *hwk1*. The homework is due prior to next week's lab, so **8/31 by 2pm**.