CS201 Homework 4

James Lindamood

October 3, 2014

1 Book Problems

1.1 Exercise 4.4

Using the definition of Big O, show that:

1. $6n^2 + 3$ is $O(n^2)$

   $6n^2 + 3 \leq 6n^2 + 3n^2$ for $n \geq 1$
   $6n^2 + 3 \leq 9n^2$ for $n \geq 1$
   Therefore,
   $6n^2 + 3$ is $O(n^2)$ with c = 9, N=1, $g(n) = n^2$

2. $n^2 + 17n + 1$ is $O(n^2)$

   $n^2 + 17n + 1 \leq n^2 + 17n^2 + 1n^2$ for $n \geq 1$
   $n^2 + 17n + 1 \leq 19n^2$ for $n \geq 1$
   Therefore,
   $n^2 + 17n + 1$ is $O(n^2)$ with c = 19, N=1, $g(n) = n^2$

3. $5n^3 + 100n^2 - n - 10$ is $O(n^3)$

   $5n^3 + 100n^2 - n - 10 \leq 5n^3 + 100n^3 - n^3 - 10n^3$ for $n \geq 1$
   $5n^3 + 100n^2 - n - 10 \leq 94n^3$ for $n \geq 1$
   Therefore,
   $5n^3 + 100n^2 - n - 10$ is $O(n^3)$ with c = 94, N = 1, $g(n) = n^3$

4. $3n^2 + 2^n$ is $O(2^n)$

   $3n^2 + 2^n \leq 2^{n+2}$ for $n > 4$
   $2^{n+2} = 4 * 2^n$
   $3n^2 + 2^n \leq 4 * (2^n)$
   Therefore,
   $3n^2 + 2^n$ is $O(2^n)$ for c = 4, N = 4, $g(n) = 2^n$

1.2 Exercise 4.9

1. Show that $7n^2 + 5n$ is not $O(n)$.

   Proof by contradiction:
   Let us assume $7n^2 + 5n \leq c * n -$ Divide by n
   $7n + 5 \leq c$
   as $n \Rightarrow \infty, c \geq \infty$ which is impossible.

1.3 Exercise 4.12

What is the order of this algorithm?

```
for(int pass = 1; pass <= n; pass++) {
  for(int index = 0; index < n; index++) {
    for(int count = 1; count < 10; count++) {
      . . . .
    }
  }
}
```

This algorithm is $O(n^2)$, as the first 2 loops require n operations in the worst case, while the final loop requires a constant of 10 or $O(1)$

1.4 Exercise 4.17

Consider four programs - A, B, C, & D. If each program requires 10 seconds to solve a problem of size 1000, estimate the time required by each program for a problem of size 2000:

$t = k * n$; $10 = k * 1000$; $k = 1/100$;

A $O(logn)$

.076 seconds.

B $O(n)$

20 seconds.

C $O(n^2)$

40,000 seconds.

D O(2^n)

1.15 * 10^{600} seconds.

1.5 Exercise 4.18

Suppose that you have a dictionary whose words are not sorted in alphabetical order. As a function of the number, n, of words, what is the time complexity of searching for a particular word in this dictionary?

The time complexity is $O(n)$ as in the worst case, the algorithm would have to search the entire array (n items) to the end to find the word.

1.6 Hydra Write-up

1. Using Big O notation, predict the time requirement for this algorithm in terms of the number n of characters in the initial string.

   I estimate that my program will take $O(n!)$ time. My program creates the child strings (from 'slaying' the larger string) by iterating over each letter of the string, which is $O(n)$ time. My program must then remove the slain head and add these child strings, which is only $O(1)$ time. Then, my program must do nearly $n!$ operations, as the initial string will break down factorialy, creating $n!$ additional strings in the array. This aspect of the program takes by far the most time, and thus is the $g(n)$ my program will trend toward.

2. Time the actual execution of the program for various values of n, and write a chart with your results. (You need not create a plot, simply a series of n / time pairs will do) For the timing, remove output statements from your program, as simply printing to the screen actually eats up a lot of time.

```
james@Kirsten:~/CS201$ time java Hydra
This is the monster you are facing: HYDRA

real    0m0.066s
user    0m0.058s
sys     0m0.009s
james@Kirsten:~/CS201$ time java Hydra 00000
This is the monster you are facing: 00000

real    0m0.066s
user    0m0.057s
sys     0m0.009s
james@Kirsten:~/CS201$ time java Hydra 000000000000
000000000000000
This is the monster you are facing: 00000000000000
0000000000000

real    0m4.316s
user    0m4.509s
sys     0m0.304s
james@Kirsten:~/CS201$ time java Hydra 000000000000
000000000000000000000000000
This is the monster you are facing: 00000000000000
0000000000000000000000000

real    293m37.079s
user    309m28.920s
sys     1m56.856s
```