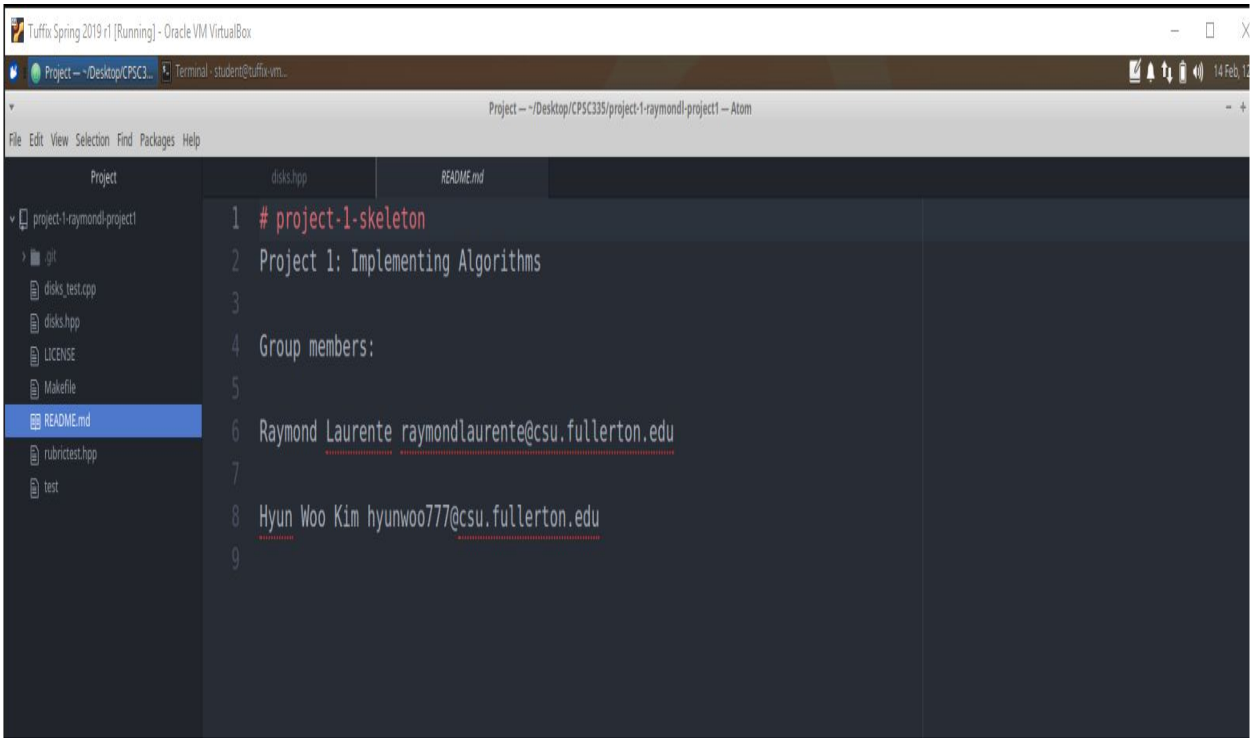
Raymond Laurente,

Hyun Woo Kim

CPSC 335 Algorithm Project 1 Written Report

This algorithm project was to determine the efficiency of (left-to-right) and lawnmower algorithm. Raymond and I had to decide what to implement and how to approach the problem. After timeless observation and thought process along with our pseudocode design we were able to complete the problem.

1. **Left-to-right**
   a. Simply left to right algorithm checks the first index of the array and checks the next index of the array and swap if necessary. Luckily, the array was in alternating order all we had to do was swap them.
2. **Lawnmower**
   a. This was the most difficult function to solve compare to the others
   b. We first divide the task in half
   c. Most outer loop will go half the size of the array
      i. First inner (left) loop will iterate half way also swapping if necessary
      ii. Second inner (right) loop will iterate half way also and swapping if necessary.
      iii. We had to make sure that the second inner loop had to be pointing the last-2
      iv. To compare the last-2 and last-1 to check statements
3. **Screenshot of READ.ME file**


Tuffix Spring 2019 r1 [Running] - Oracle VM VirtualBox
Project – ~/Desktop/CPSC3...
Terminal - student@tuffix-vm...
14 Feb, 12
Project – ~/Desktop/CPSC335/project-1-raymondl-project1 – Atom
File Edit View Selection Find Packages Help
Project
disks.hpp
README.md
project-1-raymondl-project1
.git
disks_test.cpp
disks.hpp
LICENSE
Makefile
README.md
rubrictest.hpp
test

```
# project-1-skeleton
Project 1: Implementing Algorithms

Group members:

Raymond Laurente raymondlaurente@csu.fullerton.edu

Hyun Woo Kim hyunwoo777@csu.fullerton.edu

```

4. Pseudocode

Left-to-right Algorithm

```
For I to end
        If(left(L) and Left+1(D) then swap
        Counter++
```

Lawnmower Algorithm

```
For I to end/2
        For left to end-1 and ++
                If(left(L) and Left+1(D) then swap
                        Counter++
        For right = size-2 to right>0 and –
                If(right(L) and right+1(D) then swap
                Counter++
```

5. Mathematical Analysis

a. Big O notation always consider the **worst-case scenario**
b. left-to-right algorithm = **O(n^2)**
c. lawnmower algorithm = **O(n^3)**
d. proof and analysis are written at the bottom

Left-to-right

```
For I = 1 to n do
    For j =1 to n do
        If(disk == D) && disk==L // 3
                Disk swap
                Swap++          //2 steps
                                // no else therefore, 0
```

// the if statement step counter

3 + max(2,0) = 5 steps

// the inner for loop

[(n-1)/1] +1 * 5 = 5n

// outer for loop

[(n-1)/1] + 1 * 5n = 5n^2

therefore, the Big(O) notation for this algorithm is **O(n^2)**

Lawn Mower Algorithm

```
for I= 0 to n do
      for left=0 to n - i;
                  if(disk(left) == L && disk(left+1) ==D
                              swap
                              counter++
      for right = n-2 to right>0
                  if(disk(right)  == L && disk(right+1) == D
                              swap
                              count++
```

// if- statement counter

3 +max(2,0) = 5

// bottom inner loop

[(n-2) - 0/1] + 1 * 5 = 5n - 5

$\sum 5n - 5$ = 5n^2 -5n

// upper inner loop

[(n-1)/1] + 1 * 5 =5n

Σ 5n = 5n^2

// most outer forl oop

[(n-1)/1)+1] * max(5n^2-5n, 5n^2)

n * 5n^2 = 5n^3

therefore, the worst case scenario for lawnmower algorithm is O(n^3)