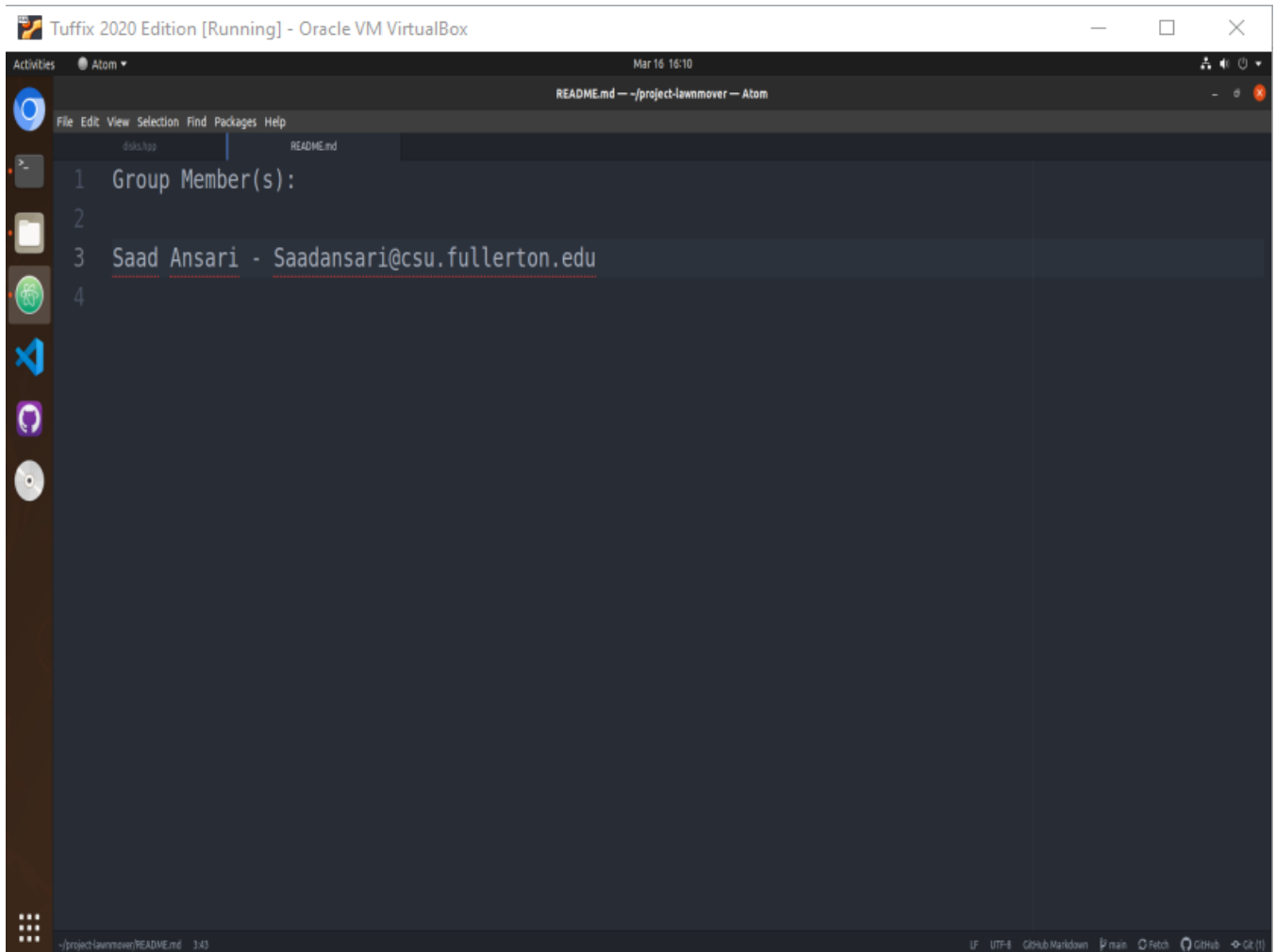


CPSC 335 Project 1

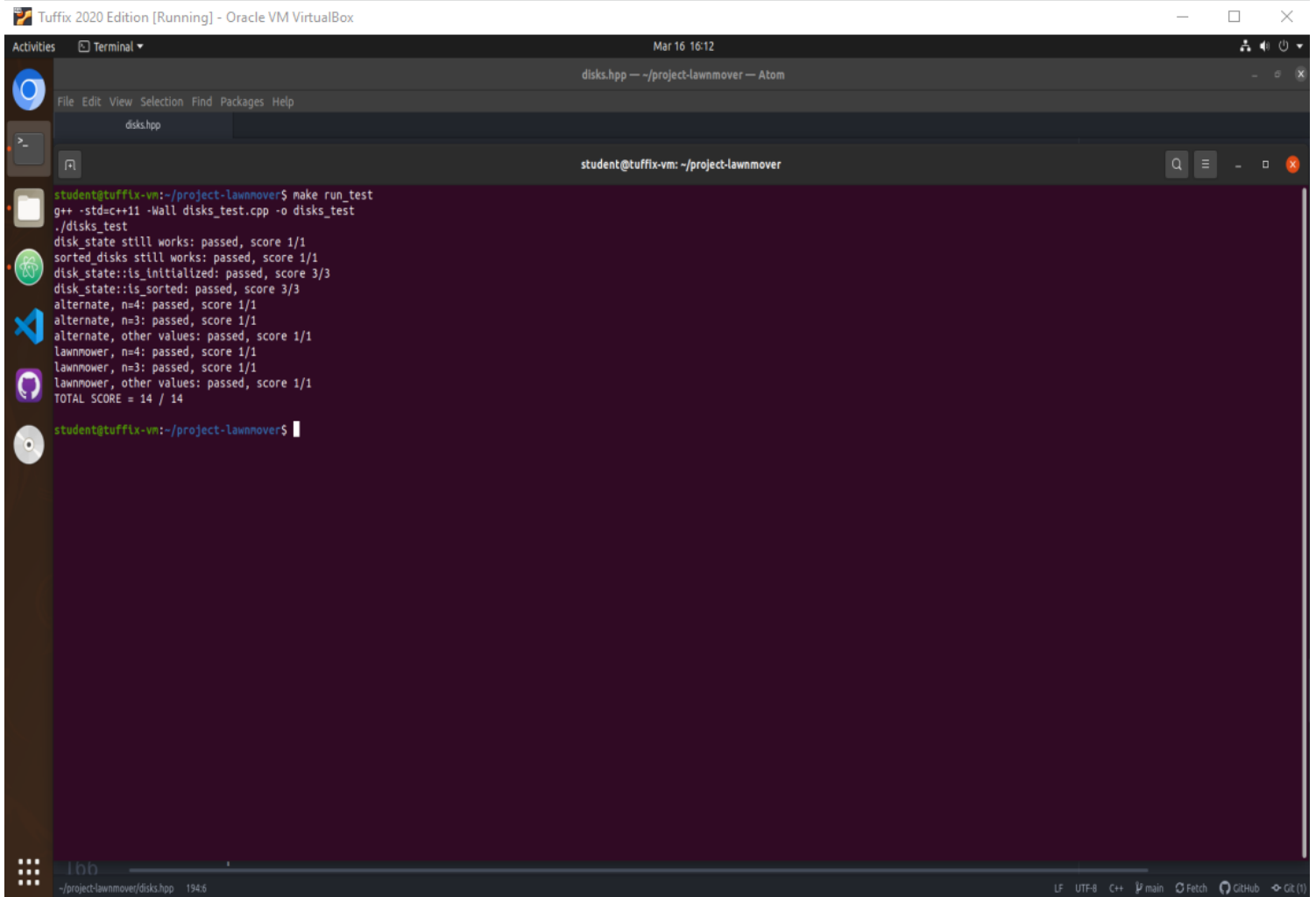
Alex Ly - aly26@csu.fullerton.edu

Saad Ansari - Saadansari@csu.fullerton.edu

README Screenshot



Code Compiling and Executing



The screenshot displays a virtual machine environment titled "Tuffix 2020 Edition [Running] - Oracle VM VirtualBox". The interface includes a top menu bar with "File", "Edit", "View", "Selection", "Find", "Packages", and "Help". Below the menu, a terminal window is open, showing the execution of a C++ program. The terminal output indicates that the program compiled successfully and passed all 14 tests, achieving a total score of 14 / 14. The tests include "disk_state still works", "sorted_disks still works", "disk_state::is_initialized", "disk_state::is_sorted", "alternate, n=4", "alternate, n=3", "alternate, other values", "lawnmower, n=4", "lawnmower, n=3", and "lawnmower, other values". The terminal prompt is "student@tuffix-vm: ~/project-lawnmover".

```
student@tuffix-vm:~/project-lawnmover$ make run_test
g++ -std=c++11 -Wall disks_test.cpp -o disks_test
./disks_test
disk_state still works: passed, score 1/1
sorted_disks still works: passed, score 1/1
disk_state::is_initialized: passed, score 3/3
disk_state::is_sorted: passed, score 3/3
alternate, n=4: passed, score 1/1
alternate, n=3: passed, score 1/1
alternate, other values: passed, score 1/1
lawnmower, n=4: passed, score 1/1
lawnmower, n=3: passed, score 1/1
lawnmower, other values: passed, score 1/1
TOTAL SCORE = 14 / 14
student@tuffix-vm:~/project-lawnmover$
```

Two Pseudocode Listings

Alternate Algo Pseudocode

```
sort_alternate(disk_state)
    number of swap = 0
    disk_state after = before
    size = after.total_count()
    for i = 0 to i < size / 2
        if i % 2 == 0
            for j = 0 to j < size - 1
                if left is dark AND right is light
                    swap
                    increment number of swap
            else
                for j = 1 to j < size - 2
                    if left is dark AND right is light
                        swap
                        increment number of swap
    return
```

Lawnmower Algo Pseudocode

```
sort_lawnmower(disk_state)
    number of swap = 0
    disk_state after = before
    size = after.total_count()
    for i = 0 to i < size / 2
        for j = 0 to j < size - 1
            if left is dark and right is light
                swap
                increment number of swap
    for i = size - 1 to i > 0
        if left is light and right is dark
            swap
            increment number of swap
    return
```

Two Algorithm Stepcount & Analysis

Alternate Algo Stepcount

Alternate Algorithm:

```

int numSwap = 0;
dist_state after = before;
size_t size = after.getTotalCount();
L1 for i = 0 to n/2 ← 3tu
    if (i % 2 == 0) ← 2tu
        L2 for j = 0 to n-1 jump 2
            if (after.get(i) == DARK && after.get(j+1) == LIGHT)
                after.swap(i);
                numOfSwap++;
            end loop L2
        else
            L3 for j = 1 to n-2, jump 2
                if (after.get(j) == DARK && after.get(j+1) == LIGHT)
                    after.swap(j);
                    numOfSwap++;
                end loop L3
    end if
end loop L1
    
```

$$L_2 \text{ StepCount: } \left(\frac{n-0}{2} + 1\right)(4) = \frac{4n-4}{2} = 2n-2$$

$$L_3 \text{ StepCount } \left(\frac{n-2-1}{2} + 1\right)(4) = \frac{2n-8-4}{2} + 1$$

$$= 2n-4-2+1$$

$$= 2n-5$$

$$L_2 = 2n-2, L_3 = 2n-5, L_2 > L_3$$

L1 StepCount:

$$\left(\frac{\frac{n}{2}-0}{1} + 1\right)(2+2n-2) = \left(\frac{n}{2} + 1\right)(2n) = n^2 + 2$$

$$\text{StepCount: } 3 + n^2 + 2$$

$$= n^2 + 5$$

$$n^2 + 5 \in O(n^2)$$

Prove $n^2 + 5 \in O(n^2)$
by Limits Test

$$\lim_{n \rightarrow \infty} \frac{n^2 + 5}{n^2} \Rightarrow \lim_{n \rightarrow \infty} \frac{n^2}{n^2} + \lim_{n \rightarrow \infty} \frac{5}{n^2}$$

$$= 1 + 0$$

$L = 1$, which ≥ 0 and constant.

Hence, $f(n) = n^2 + 5 \in O(n^2)$

Lawnmower Algo Stepcount

Lawnmower Algorithm

Size = n

```
int numof = 0;           1 tu
disk_state after = before 1 tu
for i = 0 to size-1
    size + size = after.totalcount() 1 tu
```

} 3 tu

L1: for i = 0 to size/2

```
L2: for j = 0 to size-1
    if (after.get(i) == Dark && after.get(i+1) == Light) { 3 tu
        after.swap(i);
        numof swap++; 1 tu
    }
end if
end loop
end loop
```

} 4

L3: for i = size-1 to i = 1; i--

```
    if (after.get(i) == Light && after.get(i-1) == Dark
        after.swap(i-1); 1 tu
        num of swap++; 1 tu
    }
end if
end loop
```

L2 Step Count

$$SC_{L2} = \left(\frac{n}{2} - 0 + 1\right) \left(\frac{n-1-0}{1} + 1\right) (4) = \left(\frac{n}{2} + 1\right)(n)(4) = \frac{4n^2}{2} + 4n = 2n^2 + 4n$$

L3 Step Count

$$SC_{L3} = \left(\frac{n-1-1}{-1} + 1\right)(2) = (n+1)(2) = 2n+2$$

$$3 + 2n^2 + 4n + 2n + 2 = 2n^2 + 6n + 5$$

Step Count $2n^2 + 6n + 5$

$O(n^2)$ proof

$2n^2 + 6n + 5 \in O(n^2)$

Limit Thm

$$\lim_{n \rightarrow \infty} \frac{f(n)}{g(n)} \rightarrow 2n^2 + 6n + 5$$

$$\lim_{n \rightarrow \infty} \frac{f(n)}{g(n)} \rightarrow n^2$$

By Limits Thm

$$\lim_{n \rightarrow \infty} \frac{2n^2 + 6n + 5}{n^2}$$

$$= \lim_{n \rightarrow \infty} \frac{2n^2}{n^2} + \lim_{n \rightarrow \infty} \frac{6n}{n^2} + \lim_{n \rightarrow \infty} \frac{5}{n^2} \quad L \geq 0$$

$$= 2 + 0 + 0 = 2$$

Hence $f(n) \in O(g(n))$

