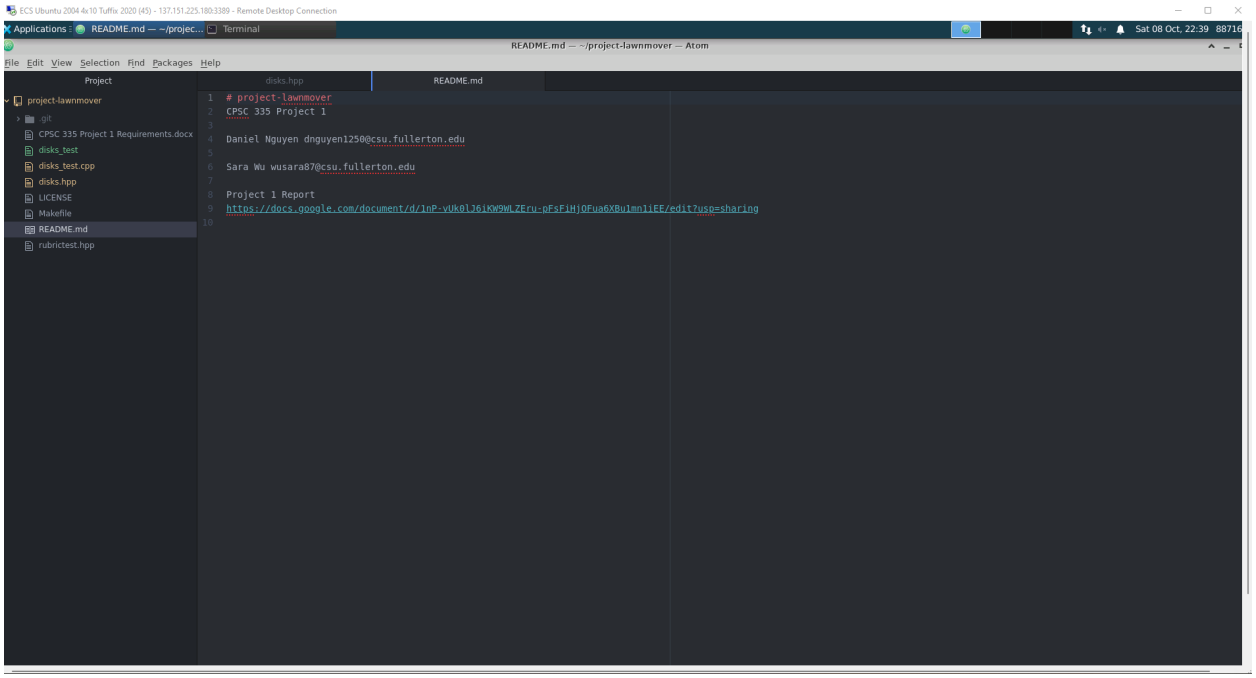
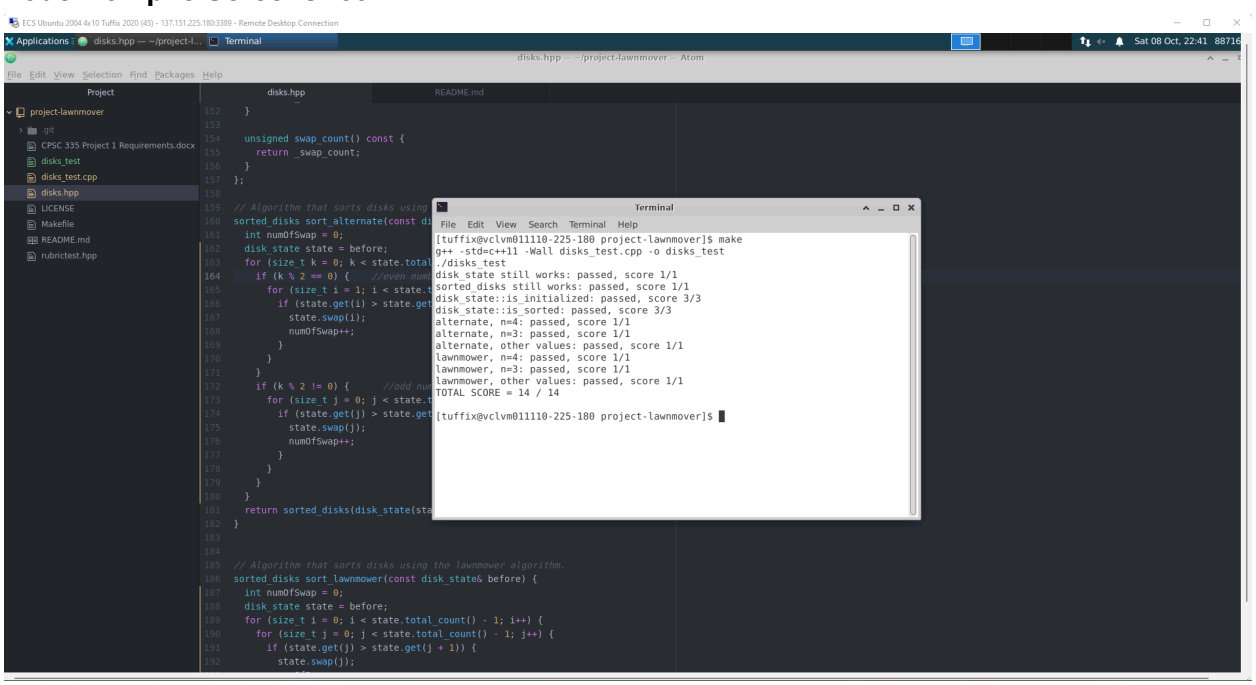
CPSC 335 Project 1 Report

Names: Sara Wu(wusara87@csu.fullerton.edu), Daniel Nguyen (dnguyen1250@csu.fullerton.edu)

README.md Screenshot



Code Compile Screenshot



The Lawnmower Algorithm

Pseudocode:

```
/**
 * Pseudocode
 * for i = 0 to 2n do
 *      swapped = false
 *      for j = 0 to 2n-1 do
 *          if L[j] > L[j+1]
 *              swap(L[j],L[j+1])
 *              increase move by 1
 *              swapped = true
 *      end for
 *      for p = 2n-1 down to 0 do
 *          if L[p] < L[p-1]
 *              swap(L[p],L[p-1])
 *              increase move by 1
 *              swapped = true
 *      end for
 *      if swapped = false then break
 * end for
 **/
```

Step Counts:

Step Counts

for i = 0 to 2n do	2n+1 times
swapped = false	1 tu
for j = 0 to 2n-1 do	2n times
if L[j] > L[j+1]	1 tu
swap(L[j], L[j+1])	1 tu
increase more by 1	1 tu
swapped = true	1 tu
~~if swapped = false then break~~	
for p = 2n-1 down to 0 do	$\overset{-2n+1}{0-(2n-1)+1} = -2n+2$ times
if L[p] < L[p-1]	1 tu
swap(L[p], L[p-1])	1 tu
increase more by 1	1 tu
swapped = true	1 tu
if swapped = false then break	1 tu

$2n*(1+\max(3,0)) = 8n$

$-2n+2*(1+\max(3,0))$

$(-2n+2)*(4)$

$-8n+8$

$1+\cancel{8n}+(-\cancel{8n}+8)+1$

$10(2n+1)$

$\boxed{20n+1 \text{ tu}}$

Proof:

$20n+1 \in O(n^2)$

$20n+1 \leq c \cdot (n^2) \ \forall n \geq n_0$

$c = 20+1 = 21 \qquad n_0 = 1$

$20n+1 \leq 21n^2$

$21 \leq 21$ ✓

By definition, $c=21, n_0=1$. This equation is true, hence $20n+1$ belongs to $O(n^2)$.

$$\lim_{n\to\infty} \frac{20n+1}{n^2} \xRightarrow[L.H.]{} \lim_{n\to\infty} \frac{20}{2n} \xRightarrow[L.H.]{} \lim_{n\to\infty} \frac{0}{2} = 0$$

By limit theorem, $L=0$. We can say that $20n+1$ does belong to $O(n^2)$.

The Alternate Algorithm

Pseudocode:

```
/*
Alternate Algo Pseudocode

for k = 0 to 2n-1 do
    if k % 2 == 0                   //even number of runs
    for  i = 1 to 2n-2 do           //(second left-most & second right-most)
        if L[i] > L[i + 1]
            swap (L[i], L[i + 1])
            numOfSwap++
            i += 2
    end for

    if k % 2 != 0                   //odd number of runs
    for  j = 1 to 2n-1 do           //(left-most & right-most)
        if L[j] > L[j + 1]
            swap (L[j], L[j + 1])
            numOfSwap++
            j += 2
    end for
    k++
end for
```

Step Counts:

```
/*
Alternate Algo Pseudocode

for k = 0 to 2n-1 do                // 2n times
    if k % 2 == 0                   // 2 tu
    for  i = 1 to 2n-2 do           // 2n - 2 times
        if L[i] > L[i + 1]          // 1 tu
            swap (L[i], L[i + 1])   // 1 tu
            numOfSwap++             // 1 tu
            i += 2                  // 1 tu
    end for

    if k % 2 != 0                   // 2 tu
    for  j = 1 to 2n-1 do           // 2n - 1 times
        if L[j] > L[j + 1]          // 1 tu
            swap (L[j], L[j + 1])   // 1 tu
            numOfSwap++             // 1 tu
            j += 2                  // 1 tu
    end for
    k++                             // 1 tu
end for
```

$(2n-2)(4) = 8n-8$

$1 + \max(3,0) = 4$

$2 + 8n - 8 = 8n - 6$

$(2n-1)(4) = 8n - 4$

$2 + 8n - 4 = 8n - 2$

$2n(8n - 6 + 8n - 2 + 1)$

$2n(16n - 7)$

$\boxed{32n^2 - 14n \text{ tu}}$

Proof:

$$32n^2 - 14n \in O(n^2)$$

$$32n^2 - 14n \leq c \cdot (n^2) \quad \forall \; n \geq n_0$$

$$c = 32 - 14 = 18 \qquad n_0 = 1$$

$$32n^2 - 14n \leq 18n^2$$

$$18 = 18 \checkmark$$

By definition, $c = 18$, $n_0 = 1$, since this equation is true, $32n^2 - 14n$ belongs in $O(n^2)$.

$$\lim_{n \to \infty} \frac{32n^2 - 14n}{n^2} \xrightarrow[L.H.]{} \lim_{n \to \infty} \frac{64n - 14}{2n} \xrightarrow[L.H.]{} \lim_{n \to \infty} \frac{64}{2} = 32$$

Since $\lim_{n \to \infty} = 32 \geq 0$, $32n^2 - 14n$ belongs in $O(n^2)$.