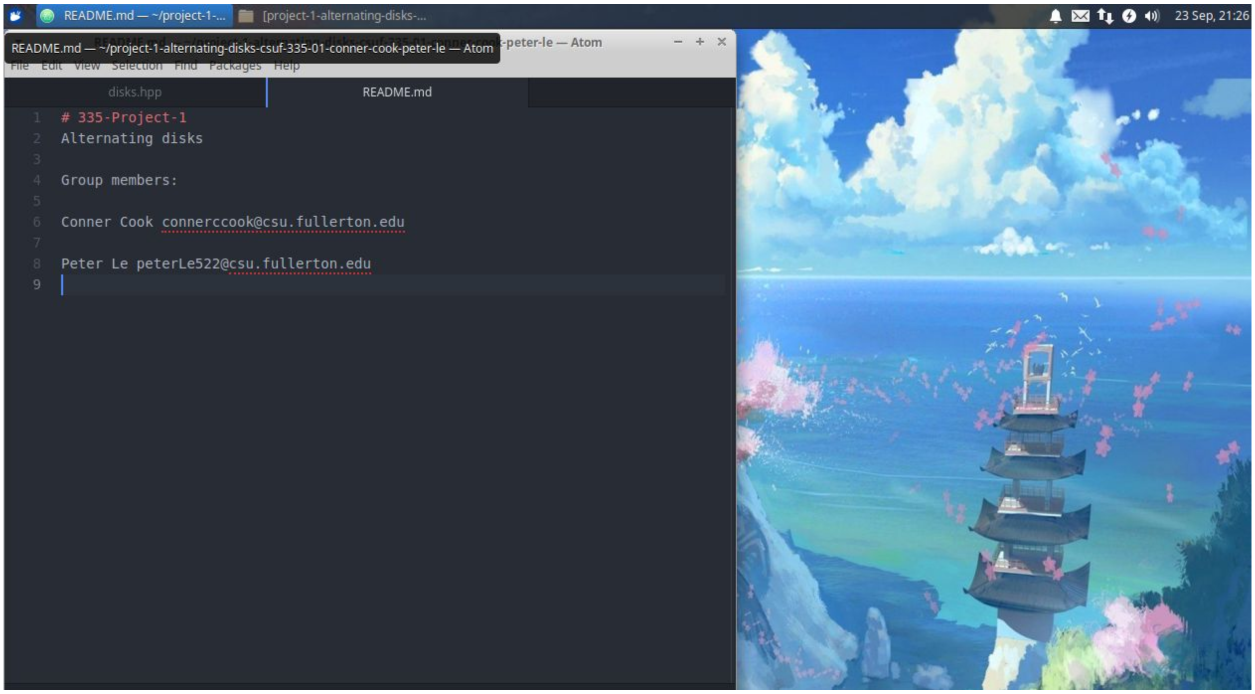
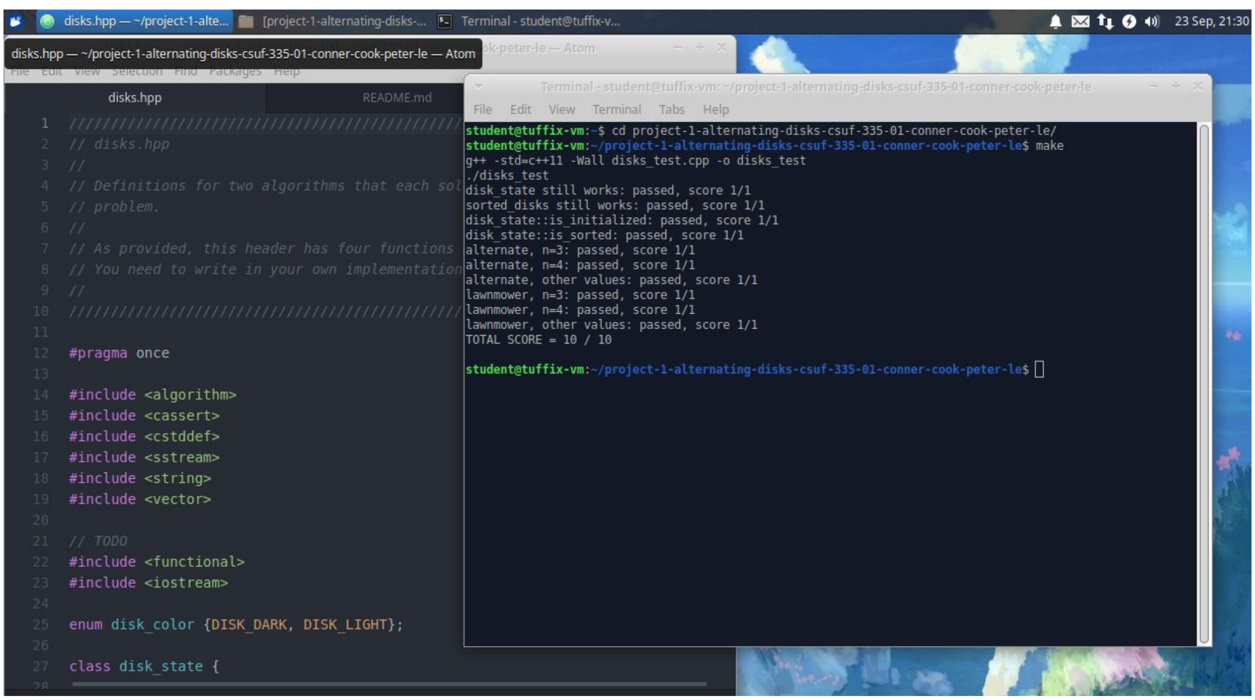
Names: Conner Cook, Peter Le
Emails: connerccook@csu.fullerton.edu, peterLe522@csu.fullerton.edu
Project #1 Alternating Disks Project Report

README.md screenshot:



Make command screenshot:



LAWNMOWER ALGORITHM

Pseudo-code:

```
counter = 0                    /*if 0 lawnmower going right, if odd going left*/-
swap_counter = 0
while disks is not sorted do
        if counter%2 == 0 then // go right
                for i = 0 to disks.length - 1 do
                        if disks[i] is light and disk[i+1] is dark then
                                swap(i)
                                swap_counter++
                endfor
                counter++
        else // /* if counter%2 is not 0, lawnmower going left */
                for i = disks.length - 1 to 0 do
                        If disks[i-1] is light and disk[i] is dark then
                                swap(i-1)
                                swap_counter++
                endfor
                counter++
endwhile
```

Mathematical Analysis:

```
counter = 0            // 1 t.u.
Swap_counter = 0   // 1 t.u.
while disks is not sorted do  //n t.u
        if counter%2 == 0 then // 2 t.u.
                for i = 0 to disks.length - 1 do // (n-1-0)+1 = n
                        if disks[i] is light and disk[i+1] is dark then // 3 t.u.
                                swap(i) 1 t.u.
                                swap_counter++ 1t.u.
                endfor
                counter++ 1 t.u.
        else // /* if counter%2 is not 0, lawnmower going left */
                for i = disks.length - 1 to 0 do // (0 - n - 1)+1 = -n
                        if disks[i-1] is light and disk[i] is dark then // 3 t.u.
                                swap(i-1) // 1 t.u.
                                swap_counter++ //1t.u.
                endfor
```

counter++ // 1 t.u.
endwhile

Step Count Calculation:

/*Tip: Read the left side top to bottom, then read the right side bottom to top */

2 + (n * whileLoopBlock) ----------------------------> 2 + (n * (5n + 3)) = $5n^2 + 3n + 2$
whileLoopBlock = 2 + max(then, else)--------------> 2 + max(5n+1, -5n+1) = 2 + 5n + 1 = 5n + 3
then = n * thenIfBlock + 1 ----------------------------> (n * 5) + 1 = 5n + 1
thenIfBlock = 3 + max(2,0) ---------------------------> 3 + 2 = 5
else = -n * elseIfBlock + 1 ---------------------------- > -n * 5 + 1 = -5n +1
elseIfBlock = 3 + max(2,0) ---------------------------> 5
Step count = $5n^2 + 3n + 2$

Definition Theorem Proof:

Show that $5n^2 + 3n + 2$ is a subset of $O(n^2)$
Have $f(n) = 5n^2 + 3n + 2$ and $g(n) = n^2$
$5n^2 + 3n + 2 <= c*n^2$ such that $c > 0$ and $n_0 >= 0$
Choose c = 10
$5n^2 + 3n + 2 <= 10n^2 \Leftrightarrow 5n^2 + 3n + 2 < = 5n^2 + 3n^2 + 2n^2$
This means that $n_0 = 1$ and $n >= 1$
Since c = 10 and c > 0 and $n_0 = 1$ and $n_0 >=0$ this means that $5n^2 + 3n + 2$ is a subset of $O(n^2)$.

Limit Theorem Proof:

Using the limit theorem, we state that $f(n) = 5n^2+3n+2$ and $g(n) = n^2$
$\lim_{x\to\infty} \ (f(n) / g(n))$
$\lim_{x\to\infty} ((5n^2 + 3n + 2) / (n^2))$
$= \lim_{x\to\infty} (5)$

By using the limit theorem, we see that the answer comes out to be a constant. This states that *f(n)* is a part of *g(n)* and *g(n)* is part of *f(n)*. Therefore, we can conclude that this algorithm is a subset of the $O(n^2)$ complexity.

The Lawnmower algorithm takes $O(n^2)$ time.

ALTERNATE ALGORITHM

Pseudo-Code:

```
iterator = 0
while disks != sorted do
        for i = 0 to disks.length do
                if (disks[i] == light && disks[i+1] == dark)
                        swap(i)
                        iterator++
                endif
        endfor
endwhile
```

Mathematical Analysis:

```
while disks != sorted do // n T.U.
        for i = 0 to disks.length-1 do // n
                if (disks[i] == light && disks[i+1] == dark) // 3 T.U.
                        swap(i) // 1 T.U.
                        iterator++ // 1 T.U.
                endif
        endfor
endwhile
```

Step Count Calculation:

$SC_{\text{While loop}}$ = # of loop executions * $SC_{\text{Loop Block}}$
= n * $SC_{\text{Loop Block}}$

$SC_{\text{Loop Block}}$= (n * $SC_{\text{if statement}}$)
= (n * 5)
= (5n)

$SC_{\text{if statement}}$ = 3 + max (2,0)
= 3 + 2
= 5

$SC_{\text{While loop}}$= n * 5n = $5n^2$

Definition Theorem Proof:

Show that $5n^2$ is a subset of $o(n^2)$

Set $f(n) = 5n^2$ and $g(n)$ to n^2

$5n^2 <= c*n^2$ such that $c > 0$ and $n0 >= 0$

Choose $c = 5$

When $c = 5$ then $5n^2 <= 5n^2$

Since the two functions are the same we can assume that n0 is 1 and $n>=1$

Therefore $5n^2$ is a subset of $O(n^2)$ which proves its efficiency class.

Limit Theorem Proof:

Using the limit theorem, we state $f(n) = 5n^2$ and $g(n) = n^2$.

$\lim_{x\to\infty} f(n) / g(n)$

$\lim_{x\to\infty} (5n^2 / n^2)$

$= \lim_{x\to\infty} (5) = 5$

By using the limit theorem, we see that the answer comes out to be a constant. This states that *f(n)* is a part of *g(n)* and *g(n)* is part of *f(n).* Therefore, we can conclude that this algorithm is a subset of the $O(n^2)$ complexity.

The alternating algorithm takes $O(n^2)$ time.