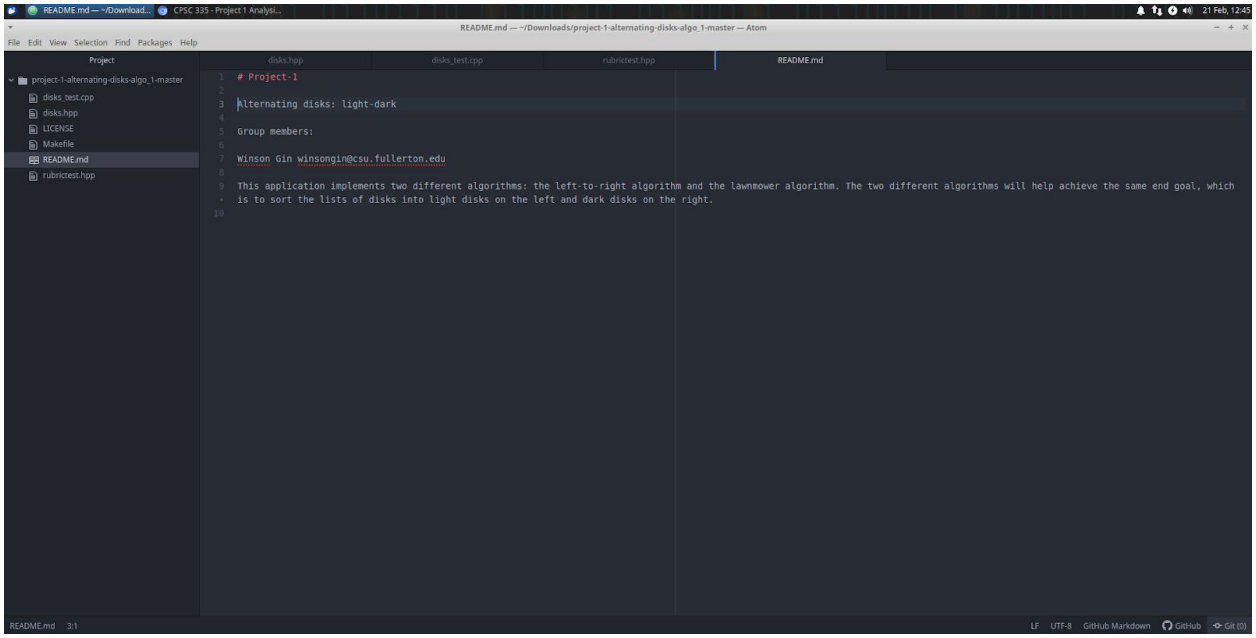
Project 1 Analysis

Group Members: Winson Gin winsongin@csu.fullerton.edu

Screenshot:



Pseudocode:

Left-to-right Algorithm

```
assert(before.is_alternating()); ---------- 7n + (9/2) times
copyOfBefore = before; ---------- 1 t.u.
swapCounter = 0; ---------- 1 t.u.
for int i = 0 to n do ---------- (n - 0/1) + 1 = n+1 times
        For i = 0 to 2n-2 do ---------- ((2n-2 - 0)/1) + 1 = 2n-1 times
                if(copyOfBefore.at(i) == DISK_DARK && copyOfBefore.at(i+1) ==
                DISK_LIGHT) ---------- 6 t.u. -------> 6 + max(5,0) = 11 t.u.
                        copyOfBefore.swap(i); ---------- 3 t.u.
                        swapCounter++; ---------- 2 t.u.
                endif
        endfor
endfor
```

return sorted_disk; ---------- 1 t.u.

Step count = (7n + 9/2) + 2 + (n+1) * (2n-1) * 11 + 1
= 7n + 15/2 + 2n^2 - n + 2n
= 2n^2 + 8n + 13/2

Lawnmower Algorithm

assert(before.is_alternating()); ---------- 7n + (9/2) times
copyOfBefore = before; ---------- 1 t.u.
swapCounter++; ---------- 1 t.u.
For i = 0 to n do ---------- n+1 times
For i = 0 to 2n-2 do ---------- (2n-2 - 0)/1 + 1 = 2n-1 times
if(copyOfBefore.at(i) == DISK_DARK && copyOfBefore.at(i+1) ==
DISK_LIGHT) ---------- 6 t.u. -------> 6 + max(5,0) = 11 t.u.
copyOfBefore.swap(i); --------- 3 t.u.
swapCounter++; ---------- 2 t.u.
endif
endfor
For i = 2n-1 to 1 do ---------- (2n-1 - 1)/1 + 1 = 2n-1 times
if(copyOfBefore.at(i) == DISK_LIGHT && copyOfBefore.at(i-1) ==
DISK_DARK) ---------- 6 t.u.--------> 6 + max(6, 0) = 12 t.u.
copyOfBefore.swap(i-1); ---------- 4 t.u.
swapCounter++; ---------- 2 t.u.
endif
endfor
endfor
return sorted_disks; ---------- 1 t.u.

Step count = (7n+9/2) + 2 + (n+1)[(2n-1)*11 + (2n-1)*12)] + 1
= (7n+9/2) + 3 + (n+1)[(22n-11) + (24n-12)] + 1
= (7n+9/2) + 3 + (n+1)[46n-23]
= 7n+9/2 + 3 + 46n^2 - 23n + 46n - 23
= 7n + 15/2 + 46n^2 -23n -23
= 46n^2 - 16n - 31/2

Proof:

Left-to-right algorithm

Step count = (7n + 9/2) + 2 + (n+1) * (2n-1) * 11 + 1
= 7n + 15/2 + 2n^2 - n + 2n
= 2n^2 + 8n + 13/2

2n^2 + 8n + 13/2 $\in \Theta(n^2)$? Yes
$\lim_{n\to\infty}$ (2n^2 + 8n + 13/2)/(n^2) = 2 >= 0

Lawnmower Algorithm

Step count = (7n+9/2) + 2 + (n+1)[(2n-1)*11 + (2n-1)*12)] + 1
= (7n+9/2) + 3 + (n+1)[(22n-11) + (24n-12)] + 1
= (7n+9/2) + 3 + (n+1)[46n-23]
= 7n+9/2 + 3 + 46n^2 - 23n + 46n - 23
= 7n + 15/2 + 46n^2 -23n -23
= 46n^2 - 16n - 31/2

46n^2 - 16n - 31/2 $\in \Theta(n^2)$? Yes
$\lim_{n\to\infty}$ (46n^2 - 16n - 31/2)/(n^2) = 46 >= 0