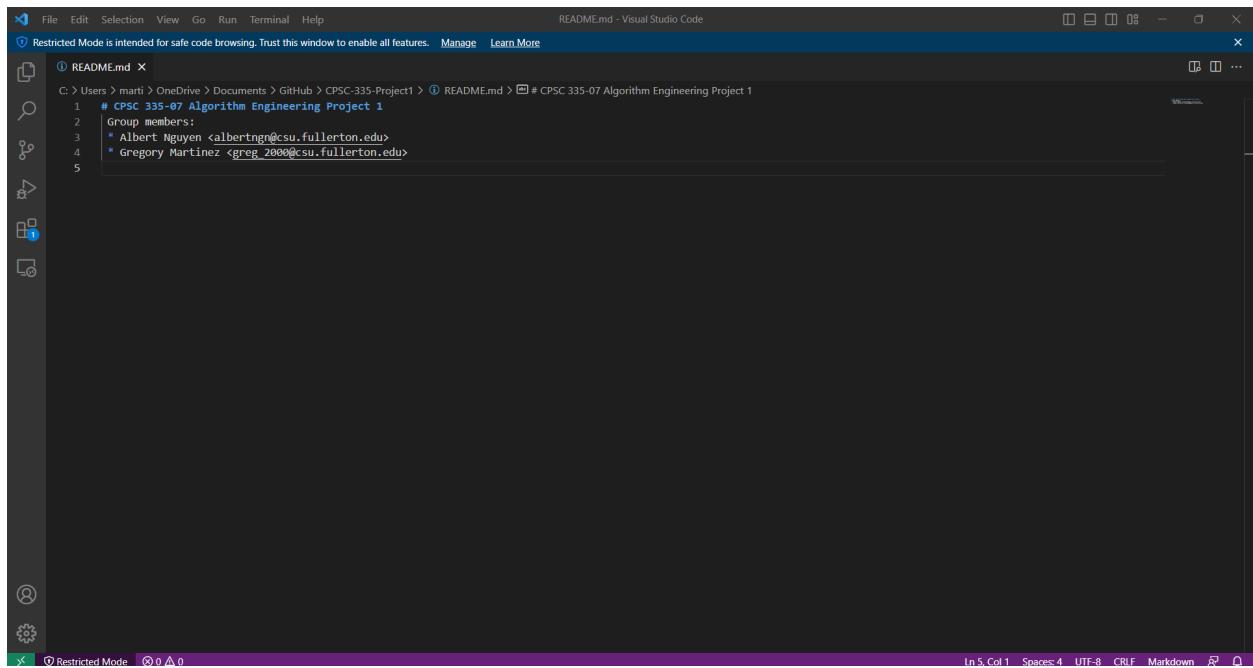


CPSC 335 Project 1: Lawnmower and Alternate Disk Algorithm

Group members:

- Albert Nguyen albertngn@csu.fullerton.edu
- Gregory Martinez greg_2000@csu.fullerton.edu

Visual Studio Code Screenshot:



The screenshot shows the Visual Studio Code interface with a dark theme. The title bar reads "README.md - Visual Studio Code". The main editor area displays the following Markdown content:

```
C: > Users > marti > OneDrive > Documents > GitHub > CPSC-335-Project1 > README.md > # CPSC 335-07 Algorithm Engineering Project 1
1  # CPSC 335-07 Algorithm Engineering Project 1
2  Group members:
3  * Albert Nguyen <albertngn@csu.fullerton.edu>
4  * Gregory Martinez <greg_2000@csu.fullerton.edu>
```

The code editor has a status bar at the bottom showing "In 5, Col 1" and other file details.

Program execution:

```

greg_2000@DESKTOP-FOJOF5L:/mnt/c/Users/marti/OneDrive/Documents/GitHub/CPSC-335-Project1$ ls
'CPSC 335 Project 1 Requirements.docx'    README.md    disks_test.cpp
LICENSE                                     disks.hpp     rubrictest.hpp
Makefile                                     disks_test  '$$CPSC 335 Project 1 Requirements.docx'
greg_2000@DESKTOP-FOJOF5L:/mnt/c/Users/marti/OneDrive/Documents/GitHub/CPSC-335-Project1$ make disks_test
g++ -std=c++11 -Wall disks_test.cpp -o disks_test
In file included from disks_test.cpp:10:
disks.hpp: In function 'sorted_disks sort_alternate(const disk_state&)':
disks.hpp:172:22: warning: comparison of integer expressions of different signedness: 'int' and 'size_t' {aka 'long unsigned int'} [-Wsign-compare]
  172 |         for(int i = 0; i < state.total_count() + 1; i++) {
                  ^~~~~~
disks.hpp:174:36: warning: comparison of integer expressions of different signedness: 'int' and 'size_t' {aka 'long unsigned int'} [-Wsign-compare]
  174 |             for(int index = 0; index < state.total_count() - 1; index += 2){
                  ^~~~~~
disks.hpp:184:34: warning: comparison of integer expressions of different signedness: 'int' and 'size_t' {aka 'long unsigned int'} [-Wsign-compare]
  184 |                 for(int index = 1; index < state.total_count() - 2; index += 2){
                  ^~~~~~
disks.hpp: In function 'sorted_disks sort_lawnmower(const disk_state&)':
disks.hpp:208:20: warning: comparison of integer expressions of different signedness: 'int' and 'size_t' {aka 'long unsigned int'} [-Wsign-compare]
  208 |             for(int i = 0; i < state.total_count() / 2; i++) {
                  ^~~~~~
In file included from disks_test.cpp:10:
disks.hpp:210:22: warning: comparison of integer expressions of different signedness: 'int' and 'size_t' {aka 'long unsigned int'} [-Wsign-compare]
  210 |                 while (index + 1 < state.total_count()) {
                  ^~~~~~
                  //iterate through all disks
greg_2000@DESKTOP-FOJOF5L:/mnt/c/Users/marti/OneDrive/Documents/GitHub/CPSC-335-Project1$ ./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
greg_2000@DESKTOP-FOJOF5L:/mnt/c/Users/marti/OneDrive/Documents/GitHub/CPSC-335-Project1$
```

Pseudocode:

```

1 | def Alternate S
2 | | numofswap = 0; ①
3 | | disk_state State [ ]; ①
4 | | for (i=0 to n+1) { ②
5 | |   if (state[i] % 2 == 0) { ③
6 | |     for (k=0 to n-1) step 2 { ④
7 | |       if (state[k] != state[k+1]) { ⑤
8 | |         if (state[k] == Dark && state[k+1] == Light) { ⑥
9 | |           swap; ⑦
10 | |           numofswap++; ⑧
11 | |         } ⑨
12 | |       } ⑩
13 | |     } ⑪
14 | |   } ⑫
15 | | else { ⑬
16 | |   for (k=1 to n-2) step 2 { ⑭
17 | |     if (state[k] != state[k+1]) { ⑮
18 | |       if (state[k] == Dark && state[k+1] == Light) { ⑯
19 | |         swap; ⑰
20 | |         numofswap++; ⑱
21 | |       } ⑲
22 | |     } ⑳
23 | |   } ㉑
24 | | } ㉒
25 | | return state []
26 | |
27 | 3

```

Alternate Step Count

$$S.C = 2 + (n+2)(S.C \text{ if/else})$$

$$S.C \text{ if/else} = 2 + \max(S.C_{\text{for}_1}, S.C_{\text{for}_2})$$

$$S.C_{\text{for}_1} = \left(\frac{n-1}{2} + 1\right) (S.C \text{ if/else})$$

↳ 2 + $\max(S.C \text{ if/else}, 0)$ = 8

↳ 4 + $\max(2, 0)$ = 6

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

$$S.C_{\text{for}_2} = \left(\frac{n-3}{2} + 1\right) (S.C \text{ if/else})$$

↳ 2 + $\max(S.C \text{ if/else}, 0)$ = 8

↳ 4 + $\max(2, 0)$ = 6

$$= \left(\frac{n-3}{2} + 1\right) (8) = 4n - 12 + 8 = \boxed{4n - 4}$$

$$\begin{aligned} S.C. &= 2 + (n+2)(2 + \max(4n+4, 4n-4)) \\ &= 2 + (n+2)(2 + 4n+4) \\ &= 2 + (n+2)(6 + 4n) \\ &= 2 + 6n + 12 + 4n^2 + 8n \\ &= \boxed{4n^2 + 14n + 14} \\ &= \boxed{O(n^2)} \end{aligned}$$

```

1 def Lawnmower Σ
2   numofswap = 0; ①
3   disk_state state [];
4   for (i=0 to n/2) Σ ② n/2
5     k=0; ①
6     while (k+1 < n) Σ n times
7       if (state[k] != state[k+1]) Σ ②
8         if (state[k] == Dark & state[k+1] == Light) Σ ④
9           swap; ①
10          numofswap++; ①
11        3
12      3
13    k++; ①
14  3
15  while (k>0) Σ k=n-1
16    if (state[k-1] != state[k]) Σ ②
17    if (state[k-1] == Dark & state[k] == Light) Σ ④
18    swap; ①
19    numofswap++; ①
20  3
21  3
22  k--; ①
23  3
24  3
25 return state [];
26 3

```

Lawnmower Step Count

$$S.C. = 2 + (n/2)(1 + S.C. \text{ while}_1 + S.C. \text{ while}_2)$$

$$\begin{aligned} S.C. \text{ while}_1 &= (n)(S.C. \text{ if/else} + 1) \\ &= (n)(2 + \max(S.C. \text{ if/else}, 0) + 1) \\ &= (n)(2 + 4 + \max(2, 0)) = 4n + 2 = 6 \end{aligned}$$

$$S.C. \text{ while}_2 = (n)(S.C. \text{ if/else} + 1)$$

$$\begin{aligned} &= (n)(2 + \max(S.C. \text{ if/else}, 0)) \\ &= (n)(2 + \max(2, 0)) = 4n + 2 = 6 \\ &= 2 + \max(6, 0) = 2 + 6 = 8 \\ &= (n-1)(8 + 1) = (n-1)(9) = 9(n-1) \end{aligned}$$

$$S.C. = 2 + (n/2)(1 + 9n - 9)$$

$$= 2 + (n/2)(18n - 18)$$

$$= 2 + (9n^2 - 4n)$$

$$= 9n^2 - 4n + 2$$

$$= O(n^2)$$

Time Complexity Proofs:

Alternate Algorithm Time Complexity Proof

$$4n^2 + 14n + 14 \in O(n^2)$$

By definition:

$$4n^2 + 14n + 14 \leq c \cdot n^2 \quad \forall n \geq n_0$$

choose $c = 32$ and $n_0 = 1$

$$\begin{aligned} 4(1)^2 + 14(1) + 14 &\leq 32 \cdot (1)^2 \\ 4 + 14 + 14 &\leq 32 \\ 32 &\leq 32 \end{aligned}$$

Hence we can say $4n^2 + 14n + 14 \in O(n^2)$.

Lawnmower Algorithm Time Complexity Proof

$$9n^2 - 4n + 2 \in O(n^2)$$

Limits theorem: $\lim_{n \rightarrow \infty} \frac{9n^2 - 4n + 2}{n^2} = \lim_{n \rightarrow \infty} \frac{\frac{9n^2}{n^2} - \frac{4n}{n^2} + \frac{2}{n^2}}{1} = 9 - 0 + 0 = 9$

By theorem, $9 \geq 0$ and is a constant. Therefore $9n^2 - 4n + 2 \in O(n^2)$.