

Project 1

CPSC 335

Aliyah Millan
and
Ethan Bockler

Emails:

millan_1802@csu.fullerton.edu

ethanbockler@csu.fullerton.edu



```
student@tuffix-vm: ~/Desktop/project-lawnmover-main$ make
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:
  TEST FAILED:
  line 81 of file disks_test.cpp, message: is_sorted() for n=3
  score 0/3
> alternate, n=4: passed, score 1/1
> alternate, n=3: passed, score 1/1
> alternate, other values: passed, score 1/1
v lawnmower, n=4:
  TEST FAILED:
  line 123 of file disks_test.cpp, message: number of swaps must be 6
  score 0/1
lawnmower, n=3:
  TEST FAILED:
  line 130 of file disks_test.cpp, message: number of swaps must be 3
  score 0/1
lawnmower, other values:
  TEST FAILED:
  line 140 of file disks_test.cpp, message: n=10 gives 45 swaps
  score 0/1
TOTAL SCORE = 8 / 14

make: *** [Makefile:13: run_test] Error 1
student@tuffix-vm: ~/Desktop/project-lawnmover-main$
```

Pseudocode Lawnmower algorithm:

number_of_swaps = 0

For i=0 to 2n do

 if(movingRight)

 For j=0 to 2n-1 do

 if(disks[j] > disks[j+1] do

 disks.swap[j]

 number_of_swaps++

 else

 For j=0 to 2n-1 do

```

    if(disks[j] > disks[j-1] do
        disks.swap[j]
        number_of_swaps++

```

Return sorted disks

Pseudocode Alternate algorithm:

```

number_of_swaps = 0
for i=0 to 2n do
    for j=0 to 2n-1 do
        if(disks[j] > disks[j+1])           -> 1tu
            disks.swap[j]                   -> 1tu
            number_of_swaps++                -> 1tu

```

return sorted disks

Pseudocode Lawnmover algorithm step count:

Inner loop(if statement):

```

if(movingRight)
    For j=0 to 2n-1 do
        if(disks[j] > disks[j+1]) do       -> 1tu
            disks.swap[j]                   -> 1tu
            number_of_swaps++               -> 1tu

    else
        For j=0 to 2n-1 do
            if(disks[j] > disks[j-1]) do     -> 1tu
                disks.swap[j]               -> 1tu

```

number_of_swaps++ -> 1tu

1+ max(2,2) = 3, both loops same

Inner loop(with for loop)

$$(2n - 1) * 3 = 6n - 3$$

Outer loop(For i=0 to 2n do)

$$\text{Sum } \{i = 0\}^{2n} 6n - 3 = 2n(6n - 3) = 12n^2 - 6n + 1$$

Proofs:

Prove using definition method that $12n^2 - 6n + 1$ belongs to $O(n^2)$:

$$12n^2 - 6n + 1 = f(n)$$

$$n^2 = g(n)$$

$$f(n) \leq c * g(n) \text{ for all } n < n_0$$

$$c = 19$$

$$n_0 = 1$$

$$7 \leq 19$$

By definition method, $12n^2 - 6n + 1$ belongs to $O(n^2)$

Prove using limit theorem that $12n^2 - 6n + 1$ belongs to $O(n^2)$:

$$\lim(12n^2 - 6n + 1 / n^2) = \lim(24n - 6 / 2n) = \lim(24/2) = \lim(12) = 12$$

By limit theorem $12n^2 - 6n + 1$ belongs to $O(n^2)$

Pseudocode Alternate Algorithm step count:

Outer loop (for i=0 to 2n do):

$$\text{Sum } 2n * 2n - 1 = 4n^2 - 2n = O(n^2)$$

Proofs:

~ Prove using definition theorem that $4n^2 - 2n$ belongs to $O(n^2)$.

$$f(n) = 4n^2 - 2n$$

$$g(n) = n^2$$

$$f(n) \leq c * g(n) \text{ for all } n < n_0$$

$$c = 1$$

$$n_0 = 1$$

$$f(n) \leq g(n)$$

~ Prove using limit theorem that $4n^2 - 2n$ belongs to $O(n^2)$.

$$\lim_{n \rightarrow \infty} (4n^2 - 2n)/(n^2) = \lim_{n \rightarrow \infty} (4n - 2)/(n) = \lim_{n \rightarrow \infty} 4/1 = \lim_{n \rightarrow \infty} 4 = 4$$

By limit theorem, $4n^2 - 2n$ belongs to $O(n^2)$.