CS208/204: Data Structures & Abstractions
Mahidol University International College

Inclass L12: Performance Characterization I: Asymptotics I

Austin J. Maddison

June 15, 2023

Problem 1 - Puzzle

1.1 Puzzle code snippet

```
def foobar(numbers: list[int]) -> list[int]:
    # make a copy of numbers
    numbers = copy(list(numbers))
    for j in range(1, len(numbers)):
        key = numbers[j]
        i = j-1
        while i >= 0 and numbers[i] > key:
            numbers[i+1] = numbers[i]
            i = i-1
        numbers[i+1] = key
    return numbers
```

Best case running time:

The best case time complexity is when `numbers[]` is sorted. The function loops through the outer loop n times. The program never enters the inner while loop as the condition is never met.

Worst case running time:

The worst case time complexity is when `numbers[]` is sorted in the opposite order, which is descending order. The reason why it is the worst is because every item in the list needs to 'bubble up' to the top of the list. The worst case time complexity is therefore $O(n^2)$ as the algorithm outer for loop has to run n times and the inner loop has to run n times. This means for each item in the `numbers[]` the algorithm has to loop through all the items in the `numbers[]` again resulting in $O(n \times n) \rightarrow O(n^2)$.

Problem 2 - Asymptotics

Problem Statement: Suppose $f(n)$ is $\Theta(s(n))$. Let $g(n) = n \cdot f(n)$. Prove that $g(n)$ is $\Theta(n \cdot s(n))$.

$f(n)$ is $\Theta(s(n))$. This means that there is positive constants c_1, c_2, and n_0 such that for all $n \geq n_0$, $f(n)$ lies between $c_1 \cdot s(n)$ and $c_2 \cdot s(n)$.

Consider the function $g(n) = n \cdot f(n)$. Want to prove that $g(n)$ is also $\Theta(n \cdot s(n))$, meaning it has the same order of growth as $n \cdot s(n)$.

To prove this, we need to show that there exist positive constants c_3, c_4, and n_1 such that for all $n \geq n_1$, $g(n)$ lies between $c_3 \cdot (n \cdot s(n))$ and $c_4 \cdot (n \cdot s(n))$.

Using the properties of Θ notation and the fact that $f(n)$ is $\Theta(s(n))$, we have:

$$c_1 \cdot s(n) \leq f(n) \leq c_2 \cdot s(n) \quad \text{for all } n \geq n_0$$

Now, let's multiply both sides of these inequalities by n:

$$c_1 \cdot (n \cdot s(n)) \leq n \cdot f(n) \leq c_2 \cdot (n \cdot s(n)) \quad \text{for all } n \geq n_0$$

Rewriting these inequalities using $g(n)$, we get:

$$c_1 \cdot (n \cdot s(n)) \leq g(n) \leq c_2 \cdot (n \cdot s(n)) \quad \text{for all } n \geq n_0$$

Thus, we can conclude that $g(n)$ is $\Theta(n \cdot s(n))$.