Worksheet 16 Solution

March 29, 2020

Question 1

a. **Part 1.a - Finding minimum possible change for a loop in a single iteration**

The minimum possible change in a loop occurs when i increments by 1.

Part 1.b - Finding maximum possible change for a loop in a single iteration

The maximum possible change in a loop occurs when i increments by 6.

Part 2.a - Determine formula for an exact lower bound on the value

Since the loop starts at $i = 0$ and ends at $n - 1$, the loop has

$$n - 1 + 1 = n \tag{1}$$

iterations.

Since the smallest step increases by 1 per iteration, the total cost of the loop at minimum possible change is

$$(n) \cdot 1 = n \tag{2}$$

steps.

Part 2.a - Determine formula for an exact upper bound on the value

Since the loop starts at $i = 0$ and ends at $n - 1$, the loop has

$$n - 1 + 1 = n \tag{3}$$

iterations.

Part 2.b - Determine formula for an exact lower bound on the value

Since the largest step increases by 6 per iteration, the total cost of the loop at minimum possible change is

$$\left\lceil \frac{n}{6} \right\rceil \tag{4}$$

steps.

Part 3.a - Determine formula for an exact upper bound on the value Is it n?

Part 3.a - Determine formula for an exact upper bound on the value Is it $\lceil \frac{n}{6} \rceil$?

Part 4 - Determine Big Oh and Big Omega

The big Oh bound of running time is $\mathcal{O}(n)$, and the big theta of running time is $\Omega(n)$.

Since n in $\mathcal{O}(n)$ and $\Omega(n)$ are the same, $\Theta(n)$ is also true.

<u>**Correct Solution:**</u>

Part 1.a - Finding minimum possible change for a loop in a single iteration

The minimum possible change in a loop occurs when i increments by 1.

Part 1.b - Finding maximum possible change for a loop in a

single iteration

The maximum possible change in a loop occurs when i increments by 6.

Part 2.a - Determine formula for an exact upper bound on the value

The upper bound of loop termination is when $k \geq n$

Part 2.b - Determine formula for an exact lower bound on the value

The lower bound of loop termination is when $6k \leq n$

Part 3.a - Use the formula to determine the exact number of loops that will occur for upper bound

Since the loop starts from 0 and ends at $n-1$, the loop has total of

$$n - 1 - 0 + 1 = n \tag{5}$$

iterations.

Since 1 step is taken for each iteration, the upper bound total cost of loop iteration is

$$n \cdot 1 = n \tag{6}$$

Since the statement on line 2 has cost of 1, the upper bound total cost of the algorithm is $n + 1$, or $\mathcal{O}(n)$.

Part 3.b - Use the formula to determine the exact number of loops that will occur for lower bound

Since the loop starts from 0 and ends at $n-1$, the loop has total of

$$n - 1 - 0 + 1 = n \tag{7}$$

iterations.

Since 6 steps are taken for each iteration, the lower bound total cost of loop iteration is

$$\left\lceil \frac{n}{6} \right\rceil \tag{8}$$

Since the statement on line 2 has cost of 1, the lower bound total cost of the algorithm is $\left\lceil \frac{n}{6} \right\rceil + 1$, or $\Omega(n)$

Part 4 - Determine Big Oh and Big Omega

The big Oh bound of running time is $\mathcal{O}(n)$, and the big theta of running time is $\Omega(n)$.

Since n in $\mathcal{O}(n)$ and $\Omega(n)$ are the same, $\Theta(n)$ is also true.

b. **Part 1.a - Finding minimum possible change for a loop in a single iteration**

The minimum possible change for a look in a single iteration is when i increases by a factor of 2

Part 1.b - Finding maximum possible change for a loop in a single iteration

The maximum possible change for a look in a single iteration is when i increases by a factor of 3

Part 2.a - Determine formula for an exact upper bound of the loop variable after k iterations

The exact upper bound of the loop variable after k iteration is $2^k \geq n$

Part 2.b - Determine formula for an exact lower bound of the loop variable after k iterations

The exact lower bound of the loop variable after k iteration is $3^k \geq n$

Part 3.a - Use the formula to determine the exact number of loops that will occur for upper bound

The upper bound of loop iteration is $\lceil \log n \rceil$, or $\mathcal{O}(\log n)$

Part 3.b - Use the formula to determine the exact number of loops that will occur for lower bound

The lower bound of loop iteration is $\lceil \log_3 n \rceil$, or $\Omega(\log n)$

Part 4 - Determine Big Oh and Big Omega

For the upper bound, we have $\mathcal{O}(\log n)$.

For the lower bound, we have $\Omega(\log n)$

Since Big Oh and Big Omega have the same value, $\Theta(\log n)$ is also true.

Question 2

a. Since **helper1** has cost of n steps, and **helper2** has cost of n^2 steps, the algorithm has total runtime of $n^2 + n$ steps, or $\Theta(n^2)$

Attempt #2:

Since **helper1** has cost of n steps, and **helper2** has cost of n^2 steps, the algorithm has total **cost** of $n^2 + n$ steps, or $\Theta(n^2)$

Notes:

- Noticed professor uses **runtime** for $\Theta(n^2)$ or $\Theta(n)$ and **cost** for the exact cost of helper functions (i.e. $n^2 + n$)

b. Assume **helper1** has running time of $\Theta(n)$ steps and **helper2** has running time of $\Theta(n^2)$.

Because the outer loop 1 runs from $i = 0$ to $\lceil \frac{n}{2} \rceil - 1$, the outer loop 1 has

$$\left\lceil \frac{n}{2} \right\rceil - 1 + 1 = \left\lceil \frac{n}{2} \right\rceil \tag{1}$$

iterations.

Since the outer loop 1 takes n steps per iteration, the outer loop 1 has total cost of $\lceil \frac{n}{2} \rceil \cdot n$ steps.

Because the outer loop 2 runs from $j = 0$ to $j = 9$, it has

$$(9 - 0 + 1) = 10 \tag{2}$$

iterations.

Since the outer loop 2 takes n^2 steps per iteration, it has total cost of $10n^2$ steps.

Since $i = 0$ and $j = 0$ each have cost of 1, the total cost of the algorithm is $\lceil \frac{n}{2} \rceil \cdot n + 10n^2 + 2$ steps or $\Theta(n^2)$.

Notes:

- Noticed professor uses the phrase **each iteration requires** n **steps for the call to helper 1** to reference helper functions in loop.
- Noticed professor did not consider $i = 0$ and $j = 0$ into total costs. Should $i = 0$ and $j = 0$ be counted towards costs? If not, how come the cost of $len(lst)$ and **return** statement are considered in Question 1.a of worksheet 15? Are there rules such as what to include and what to omit when considering the statements with constant time?

c. Assume **helper1** function has runtime of $\Theta(n)$, and **helper2** function has runtime of $\Theta(n^2)$.

Since loop 1 runs from $i = 0$ to $n - 1$, the loop has

$$n - 1 - 0 + 1 = n \tag{1}$$

iterations.

Then, since each iteration of loop 1 requires n steps for the call to **helper1**, the loop has total cost of

$$n \cdot n = n^2 \tag{2}$$

steps.

Because we know the loop 2 runs from $j = 0$ to $j = 9$, we can conclude the loop has

$$9 - 0 + 1 = 10 \tag{3}$$

iterations.

Since each iteration of loop 2 requires n^2 steps for the call to **helper2**, the loop has total cost of

$$10 \cdot n^2 = 10n^2 \tag{4}$$

steps.

Since $i = 0$ and $j = 0$ each have cost of 1, the total cost of algorithm is

$$n^2 + 10n^2 + 2 = 11n^2 + 2 \tag{5}$$

steps, or $\Theta(n^2)$.

Correct Solution:

Let $n \in \mathbb{N}$. Assume **helper1** function has runtime of $\Theta(n)$, and **helper2** function has runtime of $\Theta(n^2)$.

Since loop 1 runs from $i = 0$ to $n - 1$ where i represents the variable for loop 1, the loop has

$$n - 1 - 0 + 1 = n \tag{1}$$

iterations.

Then, since each iteration of loop 1 requires i steps for the call to **helper1**, the loop has total cost of

$$\sum_{i=0}^{n-1} i = \frac{n(n-1)}{2} \tag{2}$$

steps.

Because we know the loop 2 runs from $j = 0$ to $j = 9$ where j represents the variable for loop 2, we can conclude the loop has

$$9 - 0 + 1 = 10 \tag{3}$$

iterations.

Since each iteration of loop 2 requires j^2 steps for the call to **helper2**, the loop has total cost of

$$\sum_{j=0}^{9} j^2 = \frac{9 \cdot (9-1)(2(9)-1)}{6} \tag{4}$$
$$= \frac{9 \cdot 8 \cdot 17}{6} \tag{5}$$
$$= 204 \tag{6}$$

steps.

Since **the statements** $i = 0$ and $j = 0$ each have cost of 1, the total cost of algorithm is

$$\frac{n(n-1)}{2} + 204 + 2 = \frac{n(n-1)}{2} + 206 \tag{7}$$

steps, or $\Theta(n^2)$.

Notes:

- Missed that the helper functions depend on loop.
- Noticed that in solutions, the variables i, j, n are assumed to be in $\mathbb{N}$. But I feel worried applying the same assumption would get me into troubles. Would marks be deducted for not mentioning about the variables n, i and j? If not, when are the times the mentioning of variables can be omitted?

Question 3

a. **Predicate Logic:** $\forall x \in \mathbb{Z}^+$, (3 loops occur) $\Rightarrow \exists x_{final} \in \mathbb{Z}^+$, and $m \in \mathbb{N}$, $x - x_{final} = 2^m$