

Toshua Km

SCIT 104

NO.
DATE

HW 7 Qs. 3-4

3.a) $i = i * i$ would take i amt of time and since the code's argument takes $\log(n)$ while $i \leq n$, we can express the inner term as $\log(n)$ and thus the while loop would cause the overall RT to be $\Theta(\log(\log(n)))$.
 $i = 2 \rightarrow i = 2 * 2 \rightarrow i = 2^2$ etc etc

b) $T(n) = \sum_{i=1}^n [\Theta(i) + O(1)] \rightarrow \text{example } n = 6$
 inner loop runs = $4^3 + 8^3 + 12^3 + 16^3$ times
 $= (1 \cdot \sqrt{6})^3 + (2 \cdot \sqrt{6})^3 + (3 \cdot \sqrt{6})^3 + (4 \cdot \sqrt{6})^3 = \sum_{k=1}^{\sqrt{n}} \Theta(k \sqrt{n})^3 = \sum_{k=1}^{\sqrt{n}} \Theta(k^3 \cdot n^{3/2}) = n^{3/2} \sum_{k=1}^{\sqrt{n}} \Theta(k^3) = n^{3/2} \cdot \Theta(n^2) \Rightarrow \Theta(n^{7/2})$

c) Nested for loop - outer = $O(n)$, inner = $O(n-i)$, for loop in conditional simply $O(n)$, Taking out the constant $i!$ gives us $O(n^2)$.

d) first for loop = $O(n)$ new size = $\frac{3 \cdot 10^{25}}{2}$ etc etc $i < n$
 if and for continue onward then new size ≥ 0
 $\sum_{i=1}^n \Theta(\log(n^2)) \rightarrow \text{simple } O(\log(n^2))$
 $q(i) = i * i = i^2 = \log(i)$