Written Homework #1

Part (a)

$$i = \overset{2^1}{2}, \overset{2^2}{4}, \overset{2^4}{16}, \overset{2^8}{256}, \ldots$$

$\rightarrow 1, 2, 4, 8 = 2^{n-1}$ (Geometric series, ratio of 2)

$\rightarrow i = 2^{2^{n-1}}$ ← closed formula

$2^{2^{k-1}} = n \rightarrow 2^{k-1} = \log(n) \rightarrow k-1 = \log(\log(n))$

$\rightarrow k = \log(\log(n)) + 1$

$$T(n) = \sum_{i=2}^{\log(\log(n))+1} \Theta(1) = \Theta(\log(\log(n)) + 1)$$

$$= \boxed{\Theta(\log(\log(n)))}$$

---

part (b)

$$T(n) = \sum_{i=1}^{n} \left(\Theta(1) + c\left(\sum_{k=0}^{i^3} \Theta(1)\right)\right)$$

Since the loop runs until i^3 at a linear rate and the work the code does is $\Theta(1)$, this is $\Theta(n^3)$. The if condition is represented by $\Theta(1)$. Note this for later

$\hookrightarrow \sum_{i=1}^{n} \Theta(1) + \sum_i \sum_{k=0}^{i^3} \Theta(1)$

Since the if condition only executes when `i % (int) sqrt(n) == 0` the summation goes from 1 to sqrt(n). If $n = 9$, the loop would execute at $i = 3, 6, 9$. This means it triggered 3 times, which is $\sqrt{9}$. So:

$\hookrightarrow T(n) = \Theta(n) + \underbrace{\sum_{i=1}^{\sqrt{n}} \Theta(n^3)}$

if condition ↗ Explained by previous two paragraphs

$\hookrightarrow T(n) = \Theta(n) + \Theta(n^3 \cdot \sqrt{n})$

$$= \Theta(n) + \Theta(n^{7/2}) = \boxed{\Theta(n^{7/2})}$$

Scanned with CamScanner

Part (c)

$\rightarrow T(n) = \sum_{i=1}^{n} \sum_{K=1}^{n} \sum_{m=1}^{n} \Theta(1)$

$m = m + m$

$m = 1, 2, 4, 8, 16, 32, 64, 128, \ldots, 2^{i}$

$\rightarrow 2^{i} = n \rightarrow i = \log_2(n)$

$\hookrightarrow \sum_{i=1}^{n} \sum_{K=1}^{n} \Theta(1)$ or $\Theta(\log_2(n))$

if condition

3rd nested loop

A = | 1 | 1 | 1 | 1 | 1 | 1 |

A = | 1 | 2 | 3 | 4 | 5 | 6 |

Both arrays would run n times, so no matter the array the if condition will execute n times. However, the if condition gets checked for every K loop iteration, and every i iteration so:

$\hookrightarrow \sum_{i=1}^{n} \sum_{K=1}^{n} \Theta(1) + \Theta(n \log(n))$

$\rightarrow \Theta(n^2) + \Theta(n \log n)$

$\hookrightarrow = \boxed{\Theta(n^2)}$

Part (d)

$$T(n) = \sum_{i=0}^{n} \left(\theta(1) + \sum_{j=0}^{\overset{n}{size}} \theta(1) \right)$$

- The first $\theta(1)$ represents all the simple computations done outside the if condition such as:
  - $a = b$, $size = newsize$, etc

- The second summation represents the inner loop which does some $\theta(1)$ calculation as well until it reaches the value of size which is some n so:

$$\rightarrow T(n) = \sum_{i=0}^{n} \theta(1) + \theta(\overset{size}{n})$$

- In general, since nothing changes the upper bound of the outer summation/loop it will just run n times, and it will only enter the if condition once so:

$$\rightarrow T(n) = \theta(n) + \theta(n) = \theta(2n) = \boxed{\theta(n)}$$

Written Homework #2: Problem 2

Question (a):

in1 = 1, 2, 3, 4   in2 = 5, 6

llrec

in1: 1, 2, 3, 4
in2: 5, 6
$\boxed{\text{llrec(in1, in2)}}$ 1 → llrec(in2, in1→next) : 1 → 5 → 2 → 6 → 3 → 4

in1: 5, 6
in2: 2, 3, 4
$\boxed{\text{llrec(in2, in1→next)}}$ 5 → llrec(in2, in1→next) : 5 → 2 → 6 → 3 → 4

in1: 2, 3, 4
in2: 6
$\boxed{\text{llrec(in2, in1→next)}}$ 2 → llrec(in2, in1→next) : 2 → 6 → 3 → 4

in1: 6
in2: 3, 4
$\boxed{\text{llrec(in2, in1→next)}}$ 6 → llrec(in2, in1→next) : 6 → 3 → 4

in1: 3, 4
in2: null
$\boxed{\text{llrec(in2, in1→next)}}$ → return 3 → 4

↳ in2 = null {
return in1
}

Answer: 1, 5, 2, 6, 3, 4

Question (b)

in1 = nullptr, in2 = 2

$\boxed{\text{llrec}}$

in1 = nullptr
in2 = 2
$\boxed{\text{llrec(in1, in2)}}$ → 2

↳ in1 = nullptr {
return in2
}

Answer: 2