Asymptotic Notation (40 points)

For each of the following statements explain if it true or false and prove your answer. The base of log is 2 unless otherwise specified, and \( \ln \) is \( \log_e \).

(a) \( \log(n^{70}) = O(\log(n^{1/2})) \)

Solution: True. In order to prove that \( \log(n^{70}) = O(\log(n^{1/2})) \), we need to show that there exists constants \( c \) and \( n_0 \) such that \( \log(n^{70}) \leq c \log(n^{1/2}) \) for all \( n > n_0 \).

Let \( c = 140 \). Then

\[
\log(n^{70}) = 70 \log(n) = 140 \cdot \frac{1}{2} \log(n) = 140 \log(n^{1/2})
\]

Thus, when we choose \( c = 140 \), we find that for all \( n \), \( \log(n^{70}) = c \log(n^{1/2}) \). This means that we can choose whatever value of \( n_0 \) that we want, and these two functions will satisfy the definition of big O.

(b) \( 2^n = \Theta(e^n) \)

Solution: False. Recall that \( 2^n = \Theta(e^n) \) if both \( 2^n = O(e^n) \) and \( 2^n = \Omega(e^n) \). Thus, in order to prove that \( 2^n \) is not \( \Theta(e^n) \), we need to show at least one of these conditions does not hold. It happens that \( 2^n \) is in fact \( O(e^n) \) (though we will not prove that here). So let’s prove by contradiction that \( 2^n \) is not \( \Omega(e^n) \).

Suppose \( 2^n = \Omega(e^n) \). That is, suppose there exists some \( c \) and \( n_0 \) such that, for all \( n > n_0 \), \( 2^n \geq ce^n \). Rearranging the terms in the inequality, we find that \( c \leq \left( \frac{2}{e} \right)^n \) for all \( n > n_0 \). But clearly this is impossible, because when \( n \) is sufficiently large \( \left( \frac{2}{e} \right)^n \) is less than \( c \). Thus we have a contradiction! \( 2^n \) cannot be \( \Omega(e^n) \).

(c) \( 1000(n \log^2 n + \frac{1}{2} n^2) = \Theta(n^2) \)
Solution: True. In order to prove that $1000(n \log^2 n + \frac{1}{2}n^2) = \Theta(n^2)$, we must prove that $1000(n \log^2 n + \frac{1}{2}n^2) = O(n^2)$ and $1000(n \log^2 n + \frac{1}{2}n^2) = \Omega(n^2)$.

We will start with big $O$. For all $n \geq 1$, we have that:

$$1000(n \log^2 n + \frac{1}{2}n^2) \leq 1000(n^2 + \frac{1}{2}n^2) = 1500n^2$$

Thus, if we choose $n_0 = 1$ and $c = 1500$, these functions satisfy the definition of big $O$.

Now we will prove big $\Omega$. For all $n \geq 1$, $n \log^2 n > 0$, so we have that:

$$1000(n \log^2 n + \frac{1}{2}n^2) \geq 1000(0 + \frac{1}{2}n^2) = 500n^2$$

Thus, if we choose $n_0 = 1$ and $c = 500$, then these functions satisfy the definition of big $\Omega$ as well.

(d) $3^n = \Theta(3^{(n-4)})$

Solution: True. Observe that:

$$3^n = 3^4 \cdot 3^{(n-4)}$$

Thus, let $c = 3^4$, and choose any value for $n_0$. Then we can use this equation to prove both big $O$ and big $\Omega$. This works because one asks for $\geq$, and the other asks for $\leq$, so an equality will work for both.

(e) $n \cos n = \Theta(n)$

Solution: False. $n \cos n$ is not $\Omega(n)$. We can prove this as follows. Suppose there were constants $c$ and $n_0$ such that $n \cos n \geq cn$ for all $n > n_0$. Now consider $\pi n_1$ for some odd $n_1 > \max(n_0, c)$. $\pi n_1 \cos(\pi n_1) = 0 < c \cdot \pi n_1$. Contradiction! Thus $n \cos n$ cannot be $\Omega(n)$.

(f) Let $f, g$ be positive functions. Then $f(n) + g(n) = \Omega(\min(f(n), g(n)))$

Solution: True. For all values of $n$, $f(n) + g(n) \geq \min(f(n), g(n))$. Therefore, if we choose $c = 1$, and any value for $n_0$, these functions will satisfy the definition of big $\Omega$.

(g) Let $f, g$ be positive functions, and let $g(n) = o(f(n))$. Then $f(n) + g(n) = \Theta(f(n))$

Solution: True. Recall the definition of little $o$: for all constants $c > 0$, there exists a constant $n_0$ such that $g(n) < cf(n)$ for all $n > n_0$.

Let $c_1$ be some constant, and let $n_1$ be the corresponding constant such that $g(n) < c_1f(n)$ for all $n > n_1$. Then the following is true for all $n > n_1$:

$$f(n) + g(n) < f(n) + c_1f(n) = (c_1 + 1)f(n)$$

If we let $n_0 = n_1$ and $c = c_1 + 1$, then we have proven big $O$. 

2
Proving big \( \Omega \) is very straightforward. Since \( g(n) \) is a positive function, we have that for all \( n \),

\[
f(n) + g(n) > f(n)
\]

Thus, choosing \( c = 1 \) and anything for \( n_0 \), we have proven big \( \Omega \).

(h) \( 2^{5 \log n} = O(n^2) \)

**Solution:** False. Observe that \( 2^{5 \log n} = (2^{\log n})^5 = n^5 \). \( n^5 \) is not \( O(n^2) \), because for any constant \( c \geq 1 \), and for any \( n > c \), we have that \( n^5 > n^3 > c \cdot n^2 \). And for any \( c < 1 \), and any \( n > 1 \), we have that \( n^5 > n^2 > c \cdot n^2 \). So for all values of \( c > 0 \), it is impossible to find a value of \( n_0 \) that satisfies the definition of big \( O \).

## 2 Recurrences (35 pts)

Solve the following recurrences, giving your answer in \( \Theta \) notation. For each of them you may assume \( T(x) = 1 \) for \( x \leq 5 \) (or if it makes the base case easier you may assume \( T(x) \) is any other constant for \( x \leq 5 \)). Justify your answer (formal proof not necessary, but recommended).

(a) \( T(n) = 3T(n - 5) \)

**Solution:** We can find the answer using the unrolling method:

\[
T(n) = 3T(n - 5)
\]
\[
= 3 \cdot 3T(n - 10)
\]
\[
= 3^2T(n - 15)
\]
\[
= 3^i T(n - 5 \cdot \left\lfloor \frac{n - 1}{5} \right\rfloor)
\]
\[
= 3^\left\lfloor \frac{n - 1}{5} \right\rfloor T(n - 5 \cdot \left\lfloor \frac{n - 1}{5} \right\rfloor)
\]

The last step works because we have reached the base case.

Based on these calculations, we can conclude that \( T(n) = \Theta(3^\left\lfloor \frac{n - 1}{5} \right\rfloor) = \Theta(3^{n/5}) \).

(b) \( T(n) = n^{2/3}T(n^{1/3}) + n \)

**Solution:** Draw the recursion tree. We first claim that the total value of each level is exactly \( n \). We will prove this by induction on the levels. For the base case, this is clearly true at the top level. So now suppose that it is true for level \( i \), and we are trying to prove it for level \( i+1 \). Clearly at each level all nodes have the same size, so suppose that at level \( i \) the value of each node is \( k \). Then by the inductive hypothesis, there must be exactly \( n/k \) nodes. Hence at level \( i+1 \) each node has size \( k^{1/3} \), and the total number of nodes at level \( i+1 \) is \( \frac{n}{k} \cdot k^{2/3} = n/k^{1/3} \).
Thus the total value at level \(i + 1\) is \(k^{1/3} \cdot (n/k^{1/3}) = n\). So we have completed the inductive step.

Since the total value at every level is exactly \(n\), it remains only to calculate the number of levels. Clearly at level \(i\), the value of each node is \(n^{1/3^i}\). The recursion will stop when this is 5 (or less). Thus we know that \(n^{1/3^i} \geq 5\). Taking logs of both sides, we get that \(\frac{1}{3^i} \log n \geq \log 5\), or equivalently that \(3^i \leq \log_5 n\). Taking logs again, we get that that \(i \leq \log_3 \log_5 n\). Thus the total number of levels is \(\Theta(\log \log n)\).

Putting this together we get that \(T(n) = \Theta(n \log \log n)\).

(c) \(T(n) = 4T(n/3) + n\)

Solution: We can use the Master Theorem to conclude that \(T(n) = \Theta(n^{\log_4 4})\).

(d) \(T(n) = 4T(n/4) + n \log_4 n\)

Solution: We can solve this by drawing out the recursion tree:

```
               n \log_4 n
                  /  \
                /    \
              /      \
             /        \
            /          \
           /            \
          /              \
         /                \
        /                  \
       /                    \
      /                      \
     /                        \
    /                          \
   /                            \
  /                              \
/                                \
\n\n```

Suppose \(n\) is an exact multiple of four. Then, as you can see, the tree will have exactly \((\log_4 n - 1)\) levels (since the base case is 4 rather than 1). At the \(i\)th level, there will be \(4^i\) nodes, each with value \(\frac{n}{4^i} \log_4 \frac{n}{4^i}\). This gives us the following solution to the recurrence

\(\text{Suppose } n \text{ is an exact multiple of four. Then, as you can see, the tree will have exactly } (\log_4 n - 1) \text{ levels (since the base case is 4 rather than 1). At the } i \text{th level, there will be } 4^i \text{ nodes, each with value } \frac{n}{4^i} \log_4 \frac{n}{4^i}. \)
relation:

\[ T(n) = \sum_{i=0}^{\log_4 n - 2} 4^i \cdot \frac{n}{4^i} \log_4\frac{n}{4^i} \]

\[ = \sum_{i=0}^{\log_4 n - 2} n \log_4\frac{n}{4^i} \]

\[ = \sum_{i=0}^{\log_4 n - 2} n \log_4 n - \log_4 4^i \]

\[ = \sum_{i=0}^{\log_4 n - 2} (\log_4 n - i) \]

\[ = \sum_{i=0}^{\log_4 n} i \]

(The last step follows by reversing the direction of the sum.) Based on the standard result that \( \sum_{i=1}^{n} i = \frac{n(n+1)}{2} \), we can conclude that \( T(n) = \Theta(n \log^2 n) \).

(e) \( T(n) = T(n - 3) + 5 \)

**Solution:** This problem is very similar to part (a), and again, we can solve it using the unrolling method:

\[ T(n) = T(n - 3) + 5 \]

\[ = (T(n - 6) + 5) + 5 \]

\[ = ((T(n - 9) + 5) + 5) + 5 \]

\[ \vdots \]

\[ = T(n - 3 \left\lfloor \frac{n-3}{3} \right\rfloor) + 5 \sum_{i=1}^{\left\lfloor \frac{n-3}{3} \right\rfloor} 5 \]

\[ = 1 + 5 \left\lfloor \frac{n - 3}{3} \right\rfloor \]

Again, the last step works because we reach the base case at \( T(n - 3 \left\lfloor \frac{n-3}{3} \right\rfloor) \); this can be shown using a similar analysis to the one in part (a).

Based on the unrolling, we can conclude that \( T(n) = \Theta(n) \).
3 Basic Proofs (25 pts)

(a) Let \( A, B, C, D \) be sets. Prove that
\[
(A \cup B) \cap (C \cup D) = (A \cap C) \cup (B \cap C) \cup (A \cap D) \cup (B \cap D)
\]

**Solution:** We can prove this using the distributive laws for sets, which say that for any sets \( X, Y, Z \):

\[
X \cup (Y \cap Z) = (X \cup Y) \cap (X \cup Z) \quad (1)
\]
\[
X \cap (Y \cup Z) = (X \cap Y) \cup (X \cap Z) \quad (2)
\]

Here is the proof:

\[
(A \cup B) \cap (C \cup D) = ((A \cup B) \cap C) \cup ((A \cup B) \cap D)
\]
\[
= (A \cap C) \cup (B \cap C) \cup (A \cap D) \cup (B \cap D)
\]

The first two steps use the second distributive law, while the third step uses the associative law for set union.

(b) There are 130 students registered for this class. Prove that there are at least 11 students who were all born in the same month.

**Solution:** Use the pigeonhole principle, or equivalently do a proof by contradiction. Suppose that there are not 11 students who were all born in the same month. Then in every month, at most 10 students were born in that month. Thus the total number of students is at most 120. This contradicts the fact that there are 130 students. Thus there must be 11 students who were all born in the same month.

(c) Prove by induction that \( \sum_{i=1}^{n} (2i - 1) = n^2 \) for all positive integers \( n \).

**Solution:** Use induction. For the base case, if \( n = 1 \) then \( \sum_{i=1}^{n} (2i - 1) = 1 = n^2 \), and thus the base case is true. For the inductive step, suppose that this is true for \( n - 1 \). Then
\[
\sum_{i=1}^{n} (2i - 1) = \sum_{i=1}^{n-1} (2i - 1) + (2n - 1) = (n - 1)^2 + 2n - 1 = n^2 - 2n + 1 + 2n - 1 = n^2
\]
as claimed.