CSC373 Worksheet 0 Solution

July 20, 2020

1. <u>Recurrence:</u> $T(n) = T(n-1) + n$

   <u>Guess:</u> $T(n) = \mathcal{O}(n^2)$.

   I need to show $T(n) \leq c \cdot n^2$.

$$\begin{align} T(n) &\leq c(n-1)^2 + n \\ &= c(n^2 - 2n + 1) + n \\ &= cn^2 - c2n + c + n \\ &\leq cn^2 - c2n + cn + n \\ &= cn^2 - cn + n \\ &\leq cn^2 - cn + cn \\ &= cn^2 \end{align}$$

   <u>Notes:</u>

   - Substitution method
     - Solves recurrences
       * Recurrence characters the running time of divide-and-conquer algorithm
     - How it works:
       1. Make a guess for the solution
       2. Use mathematical induction to prove the guess is correct or incorrect.

     <u>Example:</u>

     <u>Recurrence:</u> $T(n) = 2T(\lfloor n/2 \rfloor) + n$

     <u>Guess:</u> $T(n) = \mathcal{O}(n \log n)$,

     We need to show $T(n) \leq cn \lg n$.

1. Assume the bound holds for all positive $m < n$, in particular $m = \lfloor n/2 \rfloor$
2. Find the upper bound of $T(m)$

   $T(\lfloor n/2 \rfloor) \leq c\lfloor n/2 \rfloor \lg(\lfloor n/2 \rfloor)$

3. Show $T(n) = 2T(\lfloor n/2 \rfloor) + n$ leads to $T(n) \leq cn \lg n$

$$
\begin{aligned}
T(n) &\leq 2(c\lfloor n/2 \rfloor \lg(\lfloor n/2 \rfloor)) + n && (8)\\
&\leq cn \lg(n/2) + n && (9)\\
&= cn \lg(n) - cn \lg 2 + n && (10)\\
&= cn \lg(n) - cn + n && (11)\\
&\leq cn \lg(n) - cn + cn && (12)\\
&\leq cn \lg(n) && (13)
\end{aligned}
$$

4. Show that the boundary holds using mathematical induction

   Doesn't have information in detail. Skipping this for now.

– Making good guess
  * Three suggestions
  1. Using recursion tree
  2. Through practice
  3. prove loose upper and lower bounds on the recurrence and then reduce the range of uncertainty

2. Recurrence: $T(n) = T(\lceil n/2 \rceil) + 1$

   Guess: $T(n) = \mathcal{O}(\lg n)$.

   I need to show $T(n) \leq c \cdot \lg n$.

$$
\begin{aligned}
T(n) &\leq c \lg(\lceil n/2 \rceil) + 1 && (1)\\
&\leq c \lg(n/2) + 1 && (2)\\
&= c(\lg n - \lg 2) + 1 && (3)\\
&= c(\lg n - 1) + 1 && (4)\\
&= c \lg n - c + 1 && (5)\\
&\leq c \lg n - c + c && (6)
\end{aligned}
$$

Correct Solution:

Recurrence: $T(n) = T(\lceil n/2 \rceil) + 1$

Guess: $T(n) = \mathcal{O}(\lg n)$.

I need to show $T(n) \leq c \cdot \lg n$.

$$
\begin{aligned}
T(n) &\leq c \lg(\lceil n/2 \rceil) + 1 && (1)\\
&\leq c \lg(n/2) + 1 && (2)\\
&= c(\lg n - \lg 2) + 1 && (3)\\
&= c(\lg n - 1) + 1 && (4)\\
&= c \lg n - c + 1 && (5)\\
&\leq c \lg n - c + c && (6)
\end{aligned}
$$

The solution holds for $c \geq 1$.

3. Recurrence: $T(n) = 2T(\lfloor n/2 \rfloor) + n$

   Guess (Upperbound): $T(n) = \mathcal{O}(n \lg n)$.

   I first need to show $T(n) \leq c \cdot n \lg n$.

$$
\begin{aligned}
T(n) &= 2T(\lfloor n/2 \rfloor) + n && (1)\\
&= 2c\lfloor n/2 \rfloor \lg \lfloor n/2 \rfloor + n && (2)\\
&\leq 2c \cdot (n/2) \lg(n/2) + n && (3)\\
&= c \cdot n(\lg n - 1) + n && (4)\\
&= cn \lg n - cn + n && (5)\\
&\leq cn \lg n - cn + cn && (6)\\
&\leq cn \lg n && (7)
\end{aligned}
$$

   The above inequality holds for $c \geq 1$.

   Guess (Lowerbound): $T(n) = \Omega(n \lg n)$.

I first need to show $d \cdot (n-2)\lg(n-2) \leq T(n)$.

$$
\begin{aligned}
T(n) &= 2T(\lfloor (n-2)/2 \rfloor) + n && (8)\\
&\geq 2d\lfloor (n-2)/2 \rfloor \lg \lfloor (n-2)/2 \rfloor + n && (9)\\
&\geq 2d \cdot ((n-2)/2) \lg((n-2)/2) + n && (10)\\
&= d \cdot (n-2)(\lg(n-2) - 1) + n && (11)\\
&= d \cdot (n-2)\lg(n-2) - d \cdot (n-2) + n && (12)\\
&\geq d \cdot (n-2)\lg(n-2) - d \cdot (n-2) + (n-2) && (13)\\
&\geq d \cdot (n-2)\lg(n-2) - d \cdot (n-2) + d \cdot (n-2) && (14)\\
&= d \cdot (n-2)\lg(n-2) && (15)
\end{aligned}
$$

The above inequality holds for $0 \leq d < 1$.

Notes:

- Both upper bound and lower bound don't need to be the same

4.3-3

> We saw that the solution of $T(n) = 2T(\lfloor n/2 \rfloor) + n$ is $O(n \lg n)$. Show that the solution of this recurrence is also $\Omega(n \lg n)$. Conclude that the solution is $\Theta(n \lg n)$.

First, we guess $\boxed{T(n) \leq cn \lg n,}$ upper bound

$$
\begin{aligned}
T(n) &\leq 2c\lfloor n/2 \rfloor \lg \lfloor n/2 \rfloor + n \\
&\leq cn \lg(n/2) + n \\
&= cn \lg n - cn \lg 2 + n \\
&= cn \lg n + (1-c)n \\
&\leq cn \lg n,
\end{aligned}
$$

where the last step holds for $c \geq 1$.

lower bound

Next, we guess $\boxed{T(n) \geq c(n+2)\lg(n+2),}$

$$
\begin{aligned}
T(n) &\geq 2c(\lfloor n/2 \rfloor + 2)(\lg(\lfloor n/2 \rfloor + 2) + n \\
&\geq 2c(n/2 - 1 + 2)(\lg(n/2 - 1 + 2) + n \\
&= 2c\frac{n+2}{2} \lg \frac{n+2}{2} + n \\
&= c(n+2)\lg(n+2) - c(n+2)\lg 2 + n \\
&= c(n+2)\lg(n+2) + (1-c)n - 2c \\
&\geq c(n+2)\lg(n+2),
\end{aligned}
$$

where the last step holds for $n \geq \frac{2c}{1-c}$, $0 \leq c < 1$.

4. Recurrence (Merge sort):

$$
T(n) = \begin{cases} \Theta(1) & \text{if } n = 1 \\ T(\lceil n/2 \rceil) + T(\lfloor n/2 \rfloor) + \Theta(n) & \text{if } n > 1 \end{cases}
$$

Guess (upper bound): $T(n) \leq c \cdot (n-2) \cdot \lg(n-2)$

$$
\begin{aligned}
T(n) &\leq c(\lceil n/2 \rceil - 2)\lg(\lceil n/2 \rceil - 2) + c(\lfloor n/2 \rfloor - 2)\lg(\lfloor n/2 \rfloor - 2) + dn && (1)\\
&= c(n/2 + 1 - 2)\lg(n/2 + 1 - 2) + c(n/2 + 1 - 2)\lg(n/2 + 1 - 2) + dn && (2)\\
&= c((n-2)/2)\lg((n-2)/2) + c((n-2)/2)\lg((n-2)/2) + dn && (3)\\
&= c(n-2)\lg((n-2)/2) + dn && (4)\\
&= c(n-2)\lg(n-2) - c(n-2) + dn && (5)\\
&= c(n-2)\lg(n-2) - (d-c)n + 2c && (6)\\
&= c(n-2)\lg(n-2) && (7)
\end{aligned}
$$

The bound holds as long as $c > d$.

Guess (lower bound): $c \cdot (n-2) \cdot \lg(n-2) \leq T(n)$

$$
\begin{aligned}
T(n) &\leq c(\lceil n/2 \rceil + 1)\lg(\lceil n/2 \rceil + 1) + c(\lfloor n/2 \rfloor + 1)\lg(\lfloor n/2 \rfloor + 1) + dn && (8)\\
&\leq c(n/2 - 1 + 1)\lg(n/2 - 1 + 1) + c(n/2 - 1 + 1)\lg(n/2 - 1 + 1) + dn && (9)\\
&= c(n/2)\lg(n/2) + c(n/2)\lg(n/2) + dn && (10)\\
&= cn\lg(n/2) + dn && (11)\\
&= cn\lg(n) - cn + dn && (12)\\
&= cn\lg(n) + (d-c)n && (13)\\
&\leq c(n-1)\lg(n-1) && (14)
\end{aligned}
$$

The bound holds as long as $d > c$, and $0 \leq c < 1$

Notes:

- the n here is asymptotically large

5. Recurrence: $T(n) = 2T(\lfloor n/2 \rfloor + 17) + n$

   Guess (upper bound): $cn\lg n$

$$
\begin{aligned}
T(n) &\le 2c(\lfloor n/2 \rfloor + 17)\lg(\lfloor n/2 \rfloor + 17) + n && (15)\\
&\le 2c((n/2) + 17)\lg((n/2) + 17) + n && (16)\\
&= 2c(n/2)\lg(n/2) + n && (17)\\
&= cn(\lg(n) - 1) + n && (18)\\
&= cn\lg(n) - cn + n && (19)\\
&\le cn\lg(n) - cn + cn && (20)\\
&= cn\lg(n) && (21)
\end{aligned}
$$

6.

$$
\begin{aligned}
T(n) &= 4T(n/3) + n && (1)\\
&\le 4c(n/3)^{\log_3 4} + n && (2)\\
&\le 4c(1/3)^{\log_3 4} n^{\log_3 4} + n && (3)\\
&\le (4/4)cn^{\log_3 4} + n && (4)\\
&\le cn^{\log_3 4} + n && (5)
\end{aligned}
$$

We cannot advance further since n in $cn^{\log_3 4} + n$ cannot be eliminated.

With the new guess $T(n) \le cn^{\log_3 4} - dn$, we have

$$
\begin{aligned}
T(n) &= 4T(n/3) + n && (6)\\
&\le 4c(n/3)^{\log_3 4} - d(n/3) + n && (7)\\
&= 4c(n/3)^{\log_3 4} - d(n/3) + n && (8)\\
&= (4/3^{\log_3 4})cn^{\log_3 4} - d(n/3) + n && (9)\\
&= (4/4)cn^{\log_3 4} - d(n/3) + n && (10)\\
&= cn^{\log_3 4} - d(n/3) + n && (11)\\
&\le cn^{\log_3 4} - d(n/3) + n && (12)\\
&\le cn^{\log_3 4} && (13)
\end{aligned}
$$

The bound holds as long as $d \ge 3$ and $c \ge 1$.

Correct Solution:

Recurrence: $T(n) = 2T(\lfloor n/2 \rfloor + 17) + n$

Guess (upper bound): $cn \lg n$

$$
\begin{aligned}
T(n) &\le 2c(\lfloor n/2 \rfloor + 17)\lg(\lfloor n/2 \rfloor + 17) + n && (14)\\
&\le 2c((n/2) + 17)\lg((n/2) + 17) + n && (15)\\
&= 2c(n/2)\lg(n/2) + n && (16)\\
&= cn(\lg(n) - 1) + n && (17)\\
&= cn\lg(n) - cn + n && (18)\\
&\le cn\lg(n) - cn + cn && (19)\\
&= cn\lg(n) && (20)
\end{aligned}
$$

$$
\begin{aligned}
T(n) &= 4T(n/3) + n && (1)\\
&\le 4c(n/3)^{\log_3 4} + n && (2)\\
&\le 4c(1/3)^{\log_3 4} n^{\log_3 4} + n && (3)\\
&\le (4/4)cn^{\log_3 4} + n && (4)\\
&\le cn^{\log_3 4} + n && (5)
\end{aligned}
$$

We cannot advance further since n in $cn^{\log_3 4} + n$ cannot be eliminated.

With the new guess $T(n) \le cn^{\log_3 4} - dn$, we have

$$
\begin{aligned}
T(n) &= 4T(n/3) + n && (6)\\
&\le 4c(n/3)^{\log_3 4} - 4d(n/3)4d(n/3) + n && (7)\\
&= 4d(n/3) = 4c(n/3)^{\log_3 4} - 4d(n/3) + n && (8)\\
&= 4d(n/3) = (4/3^{\log_3 4})cn^{\log_3 4} - 4d(n/3) + n && (9)\\
&= (4/4)cn^{\log_3 4} - 4d(n/3) + n && (10)\\
&= cn^{\log_3 4} - 4d(n/3) + n && (11)\\
&\le cn^{\log_3 4} - 4d(n/3) + n && (12)\\
&\le cn^{\log_3 4} - 4d(n/2) + n && (13)\\
&\le cn^{\log_3 4} - 2dn + n && (14)\\
&\le cn^{\log_3 4} - 2dn + dn && (15)\\
&\le cn^{\log_3 4} - dn && (16)
\end{aligned}
$$

7. I need to show $T(n) \le cn^2$

$$\begin{aligned}
T(n) &= 4T(n/2) + n && (17)\\
&\le 4c(n/2)^2 + n && (18)\\
&= (4/4)cn^2 + n && (19)\\
&= cn^2 + n && (20)
\end{aligned}$$

We cannot advance further since n in $cn^2 + n$ cannot be eliminated.

But with the new guess $T(n) \le cn^2 - dn$, we have

$$\begin{aligned}
T(n) &= 4T(n/2) + n && (21)\\
&\le 4c(n/2)^2 - 4d(n/2) + n && (22)\\
&= (4/4)cn^2 - 2dn + n && (23)\\
&\le cn^2 - 2dn + dn && (24)\\
&= cn^2 - dn && (25)
\end{aligned}$$

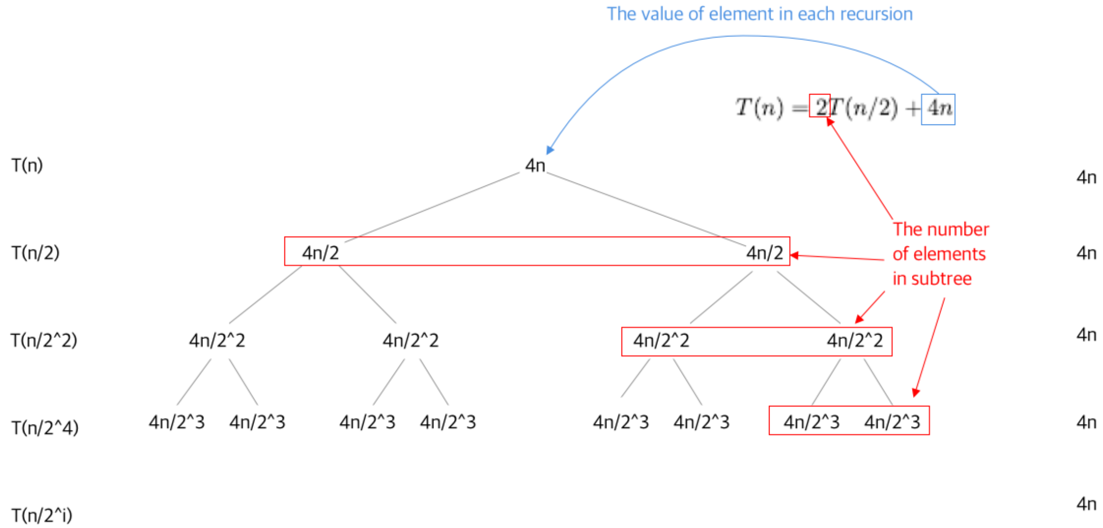
The bound holds as long as $d \ge 1$ and $c \ge 1$.

8. **Notes:**

- Recursion Tree
  - Provides a straightforward way to provide a good guess.
  - Is then verified using subtitution method

Example:

Recurrence: $T(n) = 2T(n/2) + 4n, T(1) = 4$



1. Finding number of levels in recursion tree

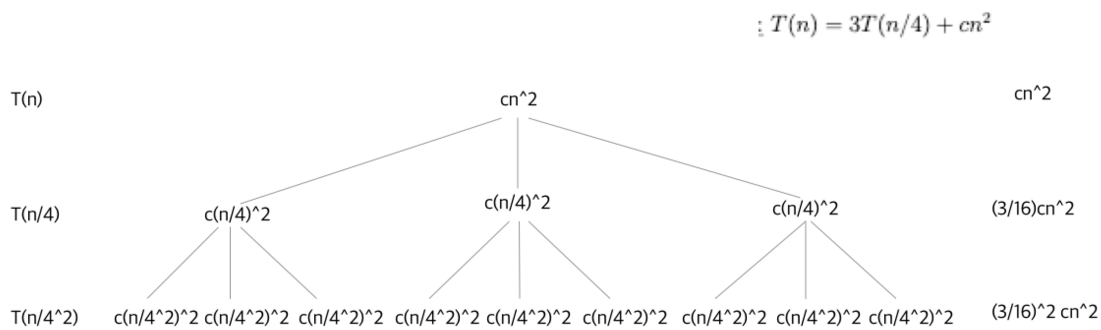
$$
\begin{aligned}
1 &= n/2^i && (1)\\
2^i &= n && (2)\\
i &= \log_2 n && (3)
\end{aligned}
$$

2. Finding the value of guess

$$
\begin{aligned}
\sum_{i=0}^{\log_2 n} 4n &= 4n \cdot \sum_{i=0}^{\log_2 n} 1 && (4)\\
&= 4n(\log_2 n + 1) && (5)
\end{aligned}
$$

Example 2:

Recurrence: $T(n) = 3T(n/4) + cn^2$



Steps:

1. Finding number of levels in recursion tree

$$
\begin{aligned}
1 &= n/4^i && (6)\\
4^i &= n && (7)\\
i &= \log_4 n && (8)
\end{aligned}
$$

2. Finding the cost of entire tree

$$
\begin{aligned}
T(n) &= \sum_{i=0}^{\log_4 n-1} c(3/16)^i n^2 + \Theta(n^{\log_4 3}) && (9)\\
&= cn^2 \cdot \sum_{i=0}^{\log_4 n-1} (3/16)^i + \Theta(n^{\log_4 3}) && (10)\\
&< cn^2 \cdot \sum_{i=0}^{\infty}(3/16)^i + \Theta(n^{\log_4 3}) \qquad \text{[since } n \text{ is asympt. large]} && (11)\\
&= cn^2\Big(\frac{1}{1-(3/16)}\Big) + \Theta(n^{\log_4 3}) \qquad \text{[Since } \sum_{i=0}^{\infty} ar^i = \frac{a}{1-r}\text{]} && (12)
\end{aligned}
$$

- **Note:** ($\log_4(n-1)$ because in $i = 0, ...i = \log_4(n-1)$ there are $\log_4(n)$ elements

3. Finding the upper bound of $T(n)$

   Since the total cost is $T(n) = cn^2\Big(\frac{1}{1-(3/16)}\Big) + \Theta(n^{\log_4 3})$, we have $\mathcal{O}(n^2)$

4. Verify the correctness of guess using subtitution method