LET T(n) DENOTE THE COST (i.e. TIME) NEEDED TO SORT THE SUB-ARRAY A[p...r], WHERE $n = r - p + 1$

THEN T(n) SATISFIES THE RECURRENCE

$$T(n) = \begin{cases} \Theta(1) & n \le 1 \\ T(\lfloor \frac{n}{2} \rfloor) + T(\lceil \frac{n}{2} \rceil) + \Theta(n) & n \ge 2 \end{cases}$$

NOTE THAT THE CONSTANTS HIDDEN BY THE Θ-NOTATION AFFECT THE SOLUTION TO THIS RECURRENCE, BUT THEY DO NOT ALTER THE <u>ASYMPTOTIC SOLUTION</u>.

WE WILL SEE IN WHAT FOLLOWS THAT THE ASYMPTOTIC SOLUTION TO THE ABOVE RECURRENCE IS:

$$T(n) = \Theta(n \log n).$$

IN MOST CASES THE ASYMPTOTIC SOLUTION WILL BE SUFFICIENT. OCCASIONLY WE WILL SEEK AN EXACT SOLUTION.

WE WILL STUDY THREE METHODS:

(4.1) SUBSTITUTION METHOD

GUESS AN ASYMPTOTIC (UPPER OR LOWER) BOUND AND TRY TO PROVE IT BY INDUCTION

(4.2) RECURSION TREE / ITERATION METHOD

SUBSTITUTE THE RECURRENCE INTO ITSELF TO OBTAIN A SUMMATION EXPRESSION WHICH CAN BE ANALYSED DIRECTLY, OR CAN BE USED TO OBTAIN A GUESS FOR THE PREVIOUS METHOD.

(4.3) MASTER METHOD

APPLY A THEOREM TO FIND ASYMPTOTIC SOLUTIONS TO RECURRENCES OF THE FORM

$$T(n) = \begin{cases} \Theta(1) & n \le n_0 \\ aT\left(\frac{n}{b}\right) + f(n) & n > n_0 \end{cases}$$

HERE $\frac{n}{b}$ REPRESENTS EITHER $\left\lfloor \frac{n}{b} \right\rfloor$ OR $\left\lceil \frac{n}{b} \right\rceil$.

4.1 Substitution

Ex. $T(n) = \begin{cases} 1 & 1 \le n < 3 \\ 2T(\lfloor n/3 \rfloor) + n & n \ge 3 \end{cases}$

Guess: $T(n) = O(n)$

We must show that there exist positive numbers c and n_0 such that:

$$(*) \quad \forall n \ge n_0 : T(n) \le cn$$

The substitution method has two parts

- Determine c and n_0 for which $(*)$ can be proved by induction. We do this by mimicing an induction proof.

- Carry out a proof of $(*)$ by induction using values for c and n_0 found in the first part.

We obtain $c = 3$, $n_0 = 1$. Show by induction that

$$\forall n \ge 1 : T(n) \le 3n.$$

PROOF:

I. $T(1) = 1 \leq 3$, SO THE BASE CASE IS SATISFIED.

II. LET $n > 1$. ASSUME FOR ALL K IN THE RANGE $1 \leq K < n$ THAT $T(K) \leq 3K$. THEN

$$\begin{aligned}
T(n) &= 2T(\lfloor n/3 \rfloor) + n \\
&\leq 2 \cdot 3\lfloor \tfrac{n}{3} \rfloor + n \qquad \text{(BY IND. HYP.)} \\
&\leq 2 \cdot 3\left(\tfrac{n}{3}\right) + n \\
&= 2n + n \\
&= 3n.
\end{aligned}$$

$\therefore T(n) \leq 3n$, AND THE RESULT FOLLOWS FOR ALL n BY THE INDUCTION PRINCIPLE. ///.

EXERCISE

SHOW THAT $T(n) = \Omega(n)$, WHENCE $T(n) = \Theta(n)$.

HINT: USE $\lfloor x \rfloor > x - 1$ FOR ANY $x \in \mathbb{R}$.

4.2 RECURSION TREES / ITERATION.

THE RECURSION TREE METHOD CAN BE USED TO GENERATE A GUESS TO BE VERIFIED BY THE SUBSTITUTION METHOD.

SINCE THIS GUESS WILL BE VERIFIED BY A MORE RIGOROUS METHOD, WE MAY TAKE SOME LIBERTIES SUCH AS DROPPING FLOORS & CEILINGS OR RESTRICTING THE VALUES OF n.

EX.
$$T(n) = \begin{cases} 1 & 1 \le n < 3 \\ 2T(\lfloor n/3 \rfloor) + n & n \ge 3 \end{cases}$$

WE ASSUME n IS AN EXACT POWER OF 3, SO THAT AT EVERY LEVEL OF THE RECURSION, $\frac{n}{3}$ IS AN INTEGER, AND $\lfloor \frac{n}{3} \rfloor = \frac{n}{3}$. THUS

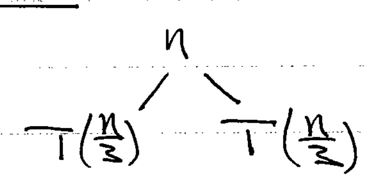
$$T(n) = 2T\left(\frac{n}{3}\right) + n$$

EACH NODE IN A RECURSION TREE REPRESENTS ONE RECURSIVE INVOCATION OF THE FUNCTION $T(\cdot)$.

0TH TREE:

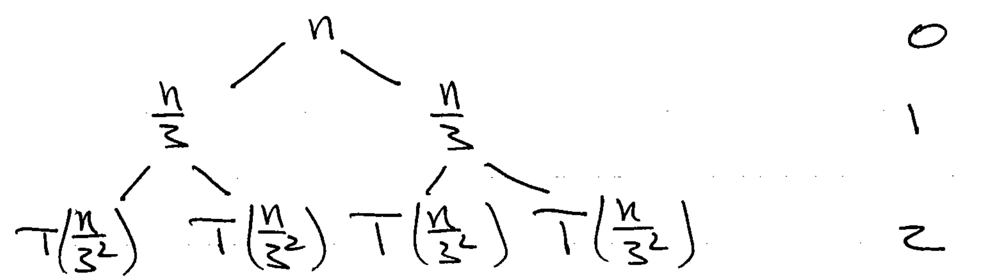
$T(n)$

1ST TREE:



12

2^{nd} TREE

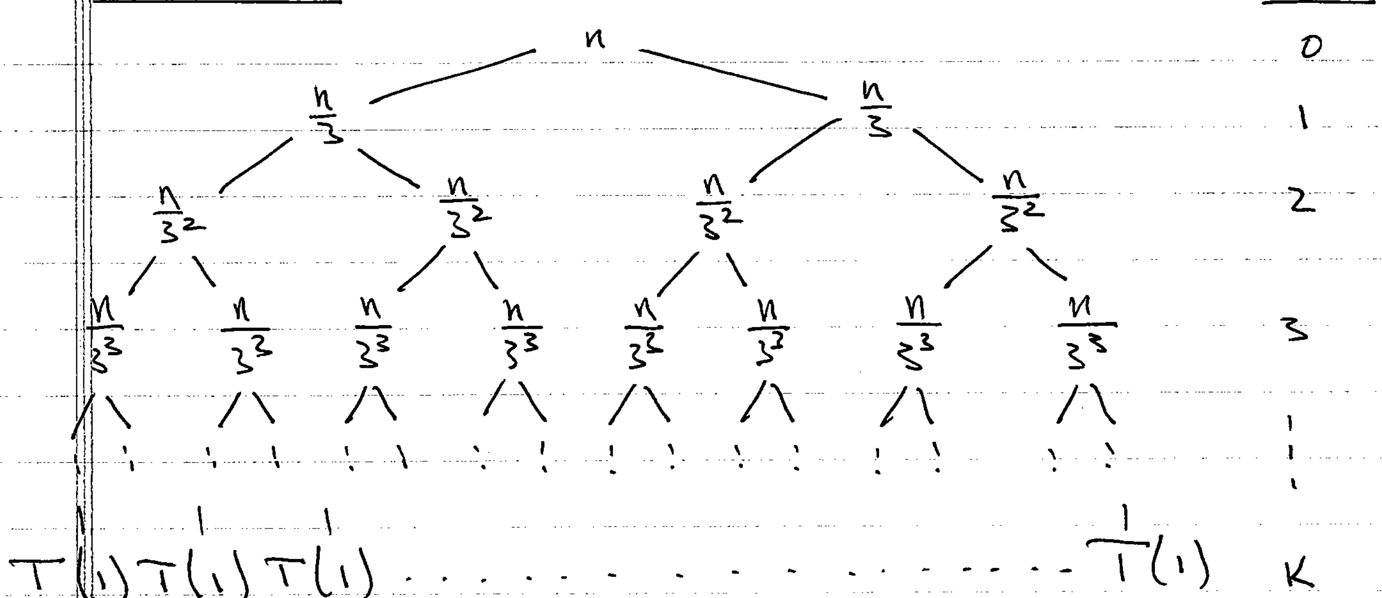


NOTE THAT $T(n)$ EQUALS THE SUM OF ALL NODES IN SUCH A TREE.

THE RECURSION CAN CONTINUE DOWN TO A LEVEL K AT WHICH ALL BOTTOM LEVEL NODES ARE $T(1)$.

K^{TH} TREE

LEVEL



K IS CHOSEN SO THAT $\frac{n}{3^K} = 1$, i.e. $K = \log_3(n)$.

FOR $0 \le i \le k-1$, THE i^{TH} LEVEL CONTAINS 2^i NODES, EACH EQUAL TO $\left(\frac{n}{3^i}\right)$. AT LEVEL k, THERE ARE 2^k NODES, EACH EQUAL TO $T(1) = 1$.

SUMMING ALL NODES ON ALL LEVELS YIELDS:

$$
\begin{aligned}
T(n) &= \sum_{i=0}^{k-1} 2^i \left(\frac{n}{3^i}\right) + 2^k \\
&= n \left(\sum_{i=0}^{k-1} \left(\frac{2}{3}\right)^i \right) + 2^{\log_3 n} \\
&= n \left(\frac{1 - \left(\frac{2}{3}\right)^k}{1 - \left(\frac{2}{3}\right)} \right) + n^{\log_3 2} \\
&= 3n \left(1 - \frac{2^{\log_3 n}}{3^{\log_3 n}} \right) + n^{\log_3 2} \\
&= 3n \left(1 - \frac{n^{\log_3 2}}{n} \right) + n^{\log_3 2} \\
&= 3n - 3 \cdot n^{\log_3 2} + n^{\log_3 2} \\
&= 3n - 2 \cdot n^{\log_3 2} \\
&= \Theta(n) \qquad (\text{SINCE } 1 > \log_3 2.)
\end{aligned}
$$

THIS CALCULATION IS VALID ONLY IF n IS AN EXACT POWER OF 3, HOWEVER IT MOTIVATES THE GUESS $T(n) = O(n)$ AND $T(n) = \Omega(n)$.