Divide-and-Conquer

This document contains summary of some of the items discussed as part of the COMS 3110 Lectures. This document **does not replace the lecture materials**. This document may contain some topics that are not covered as part of the lecture; you will not be tested on those parts, they are made available to you for gaining further knowledge on topics/concepts that are related to class-lecture.

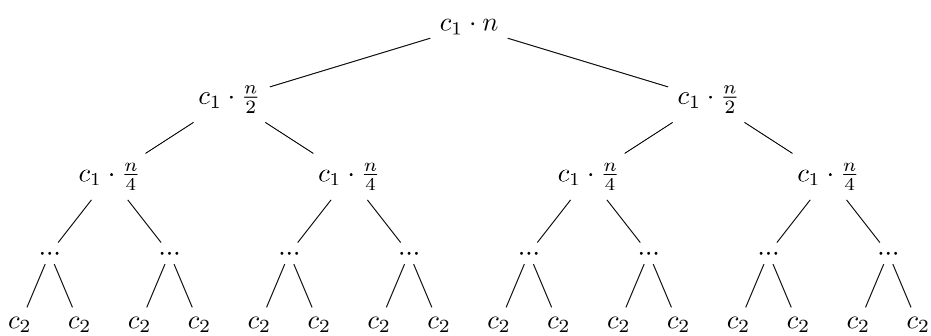
1 Topics Covered

- Divide-and-conquer: merge sort
- Recurrence tree method

2 Recurrence Tree Method on Merge Sort

The recurrence relation for Merge Sort is as follows:

$$T(n) = \begin{cases} c_2 & \text{if } n \leq 1 \\ 2 \cdot T(\frac{n}{2}) + c_1 \cdot n & \text{if } n > 1 \end{cases}$$



Level	Problem Size	# Nodes	Work / Node	Level Total
1	$\frac{n}{2^0}$	$2^0 = 1$	$c_1 \cdot n$	$c_1 \cdot n$
2	$\frac{n}{2^1}$	2^1	$c_1 \cdot \frac{n}{2}$	$c_1 \cdot n$
3	$\frac{n}{2^2}$	2^2	$c_1 \cdot \frac{n}{4}$	$c_1 \cdot n$
...	...	...	...	...
i	$\frac{n}{2^{i-1}}$	2^{i-1}	$c_1 \cdot \frac{n}{2^{i=1}}$	$c_1 \cdot n$

Assuming at level l when problem size = 1, then

$$\frac{n}{2^{l-1}} = 1 \Rightarrow l = \log(n) + 1$$

So for all i from 1 to $\log(n)$, the work done at level i is $c_1 \cdot n$. At level $\log(n) + 1$, the work done at each node is c_2, and there are $2^{\log(n)+1-1} = n$ nodes, so total work at level $\log(n) + 1$ is $c_2 \cdot n$.

$$\begin{aligned} \text{Total cost} &= \left[\sum_{i=1}^{\log n} c_1 \cdot n\right] + c_2 \cdot n \\ &= c_1 \cdot n \log n + c_2 \cdot n \\ &\in O(n \log n) \end{aligned}$$

3 Exercise

1. $T(n) = 3T(n/3) + \log n$ with $T(3) = 2$
2. $T(n) = 4T(n/2) + n^2$ with $T(2) = 4$
3. $T(n) = 2T(n/2) + 1$ with $T(1) = 0$
4. $T(n) = 8T(n/2) + n^3$ with $T(1) = 1$
5. $T(n) = T(n/2) + \sqrt{n}$ with $T(1) = 1$
6. $T(n) = 2T(n/2) + n\sqrt{n}$ with $T(2) = 2$
7. $T(n) = 5T(n/5) + n$ with $T(5) = 1$
8. $T(n) = T(n/2) + T(n/4) + n$ with $T(1) = 1$
9. $T(n) = 2T(n/3) + n$ with $T(3) = 1$
10. $T(n) = 2T(n/4) + n$ with $T(1) = 1$