Analyzing Divide-and-Conquer Algs

Reminder

\[ T(n) = aT\left(\frac{n}{b}\right) + f(n) \]

- total time for \( k \) divide and combine steps together
- \# statements (time) when input is size \( n \)
- \# of subproblems that we recursively solved
- size of each subproblem (really, \( \lfloor n/b \rfloor \) or \( \lceil \sqrt[n/b] \rceil \))

Base: \( T(1) = \Theta(1) \)

problem size where you no longer recurse
Merge sort
\[ a = 2, \ b = 2, \ f(n) = \Theta(n) \]
\[ T(n) = 2T\left(\frac{n}{2}\right) + \Theta(n) \]

Divide-and-conquer closest pair
\[ a = 2, \ b = 2, \ f(n) = \Theta(n) \]
Both divide & combine steps take linear time
Binary Search

Problem: Given a sorted array and a value \( x \), asked if \( x \) is in the array.

\[ P \quad \theta-1 \quad \theta \quad \theta+1 \quad R \]

\[ x < y \quad x > y \]

\[ T(n) = T\left(\frac{n}{2}\right) + \Theta(1) \]

\[ a = 1, \quad b = 2 \]
Strassen Matrix Multiplication

\[
\begin{bmatrix}
\begin{array}{c}
\vdots \\
\vdots \\
\vdots \\
\vdots \\
\vdots \\
\end{array}
\end{bmatrix}_{n/2} \times \begin{bmatrix}
\begin{array}{c}
\vdots \\
\vdots \\
\vdots \\
\vdots \\
\end{array}
\end{bmatrix}_{n/2} = \begin{bmatrix}
\begin{array}{c}
\vdots \\
\vdots \\
\vdots \\
\vdots \\
\vdots \\
\end{array}
\end{bmatrix}_{n/2} \times \begin{bmatrix}
\begin{array}{c}
\vdots \\
\vdots \\
\vdots \\
\vdots \\
\end{array}
\end{bmatrix}_{n/2}
\]

There are 7 multiplications of \( n/2 \) by \( n/2 \) matrices which can be combined in \( \Theta(n^2) \) time to get the desired product

\[
T(n) = 7T(n/2) + \Theta(n^2)
\]

\( a = 7, \ b = 2 \)