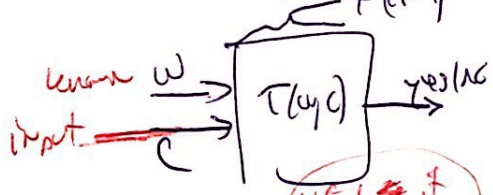
Obs Given a certificate checker program $T(w,c)$ and a polynomial time bound $f(|w|)$, there is a Boolean circuit that computes $T(w,c)$ that's polynomial in size relative to $|w|$. [duplicate a one clock-cycle computation $f(|w|)$ times].

reduction maps w to $T(w)$ (other T)

- it knows w and can build a circuit to run $T(w,c)$ for $f(|w|)$ clock cycles.



Then show $w \in L$ iff $T(w) \in 3\text{-SAT}$

iff $\exists c$ $T(w,c)$ output yes

iff $w \in L$ because $L \in NP$

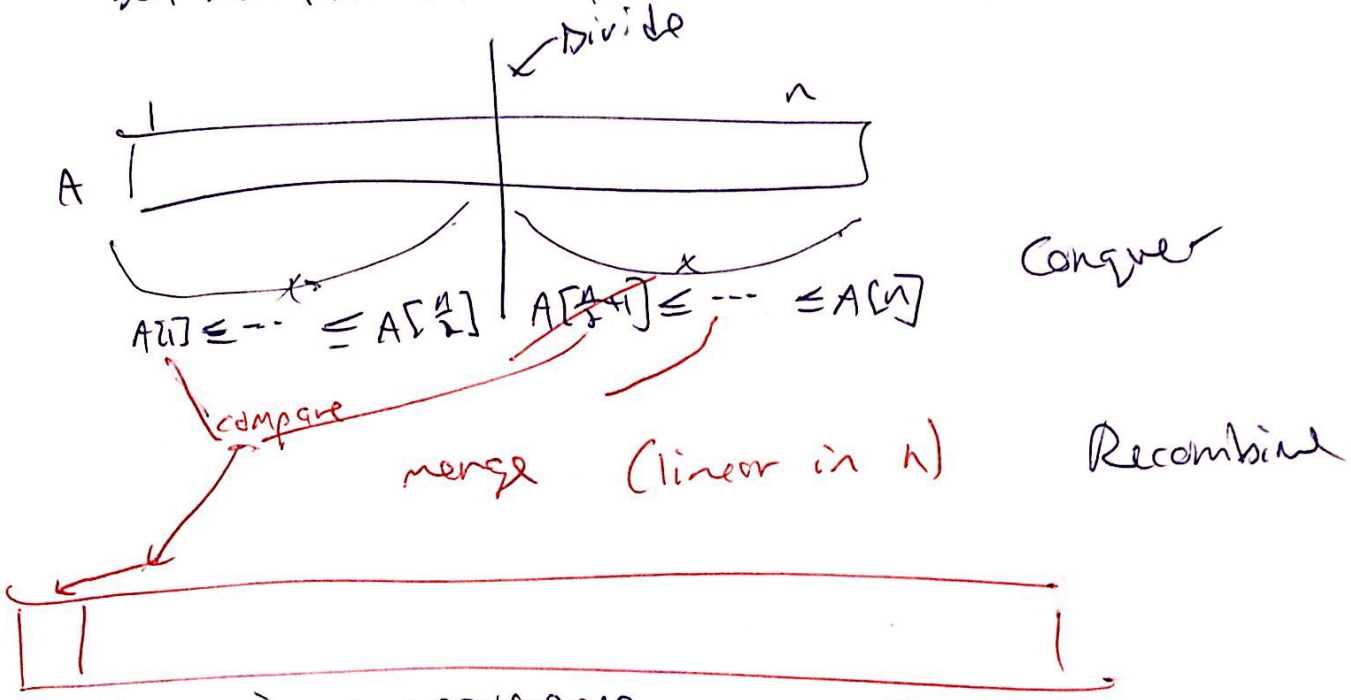
Divide and Conquer Algorithms

- typically recursive
- divide original problem into 1 or more smaller problems of the same form
  - use recursion to conquer all smaller problems
- recombine answers to answer original problem

example Sorting. $A[1..n]$ n integers

desire to rearrange so that $A[1] \leq A[2] \leq \cdots \leq A[n]$

but maintain same collection in $A[1..n]$



analyze runtime using a recurrence

recursive calls

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

merge

$$T(1) = 1$$

$$T(n) = c' n \lg n$$

Dynamic Programming

= Divide-and-conquer optimization

with memoization.

make a set of choices so that the result optimizes some scoring function