Divide & Conquer:

- Solve a # of smaller subproblems
- Combine

Merge Sort:

- Split data set in two
- Sort & combine recursively
- Merge results

\[ M(n) = 2 M(\frac{n}{2}) + O(n) \]

- \( O(n \log n) \) work per level
- \( O(n \log n) \) recurrence

Recommendation for computing resources
Dynamic Progr.
- keep table of sub-problems
  - recurrence of current problem
    i.e. previously solve problems

Pick one over and over again

\[ x_1, \ldots, x_n \]

\[ \text{LCS } X = a, b, c, a, b \]

\[ y = b, c, b, d, a \]

\[ y_{11}, \ldots, y_{nn} \]

\[ T(i, j) \text{ longest common subsequence} \]

\[ \begin{align*}
   & x_1, \ldots, x_i \\
   & y_1, \ldots, y_j
\end{align*} \]

\[ \text{repeated} \]

\[ \text{not much} \]
\[ T(i, j) = T(i-1, j-1) + 1 \quad \text{if} \quad i, j > 0 \land x_i = y_j \]

\[ T(i, j) = \max(T(i, j-1), T(i-1, j)) \quad \text{if} \quad i, j > 0 \land x_i \neq y_j \]

\[ T(0, 0) = 0, \quad i = 0, \quad j = 0 \]

- What table
- Recurrence
- Initialization
- Rec. & fill in
KnapSack

Items $s_1, \ldots, s_n$
Does there exist a knapsack of size $k$
Is subset of $s_1, \ldots, s_n$ which sums to $k$

Weakly NP-complete

$T(i, z) = \text{true}$
if $\exists$ subset of $s_1, \ldots, s_i$ summing to $z$

$T(i, z) = \begin{cases} 
T(i, z-s_i) \lor T(i-1, z) & \text{if } i \geq 1, z \geq 1 \\
F & \text{if } z = 0, i \geq 1 \\
T & \text{if } i = 0, z \geq 0 
\end{cases}$

\[
\begin{array}{c c c}
0 & \ldots & k \\
0 & T & T \\
1 & F & \text{Diagram}
\end{array}
\]