level zero

has two sons

has none
Given \( A, B, C, D \), Code into binary sequences.
1100010

C A B

A B D C

0 1 0 1
Given a set of letters, how to code optimally?
Intuitively, we use short sequences to represent commonly used letters, and long sequences to represent rarely used letters like $Q$ and $Z$.

\[
\begin{array}{cccc}
A & B & C & D \\
40\% & 30\% & 20\% & 10\%
\end{array}
\]
Huffman’s Problem

Given a set of nodes with weight \( w_j \),

find a binary tree with

\[
\sum_i w_i l_i \text{ minimized,}
\]

where \( l_j \) is the level of the node in the tree.
\[ \sum_{i} w_i l_i = 19 \]
\[
\sum_{i} w_i l_i = 25
\]
A different way to calculate $\sum_i w_i l_i$.

The weight of a father = the weight of his two sons.

The sum of $n - 1$ fathers = $\sum_i w_i l_i$. 
Necessary condition for optimality.

Never give a large $w_i$ a lower level and a small $w_j$ a higher level.
Give the two smallest weights the same father.

\[ w_1 < w_2 < \cdots < w_n \]

Given \( w_1 \) and \( w_2 \) have the same father.