Example: Huffman Trees

Idea:
Given a set of \emph{n weights}, find a binary tree with \emph{n leaves} that best stores these weights in the sense that bigger weights should be closer to the root.

\textbf{Definition:} A binary tree is a \textit{Huffman} tree if:

1. Each non-leaf vertex has exactly two children.
2. Each leaf has a weight (real number) in the range 0 to 1; the sum of all leaf weights is 1.
3. The sum over all the leaves of the leaf's depth times its weight is minimum among all binary trees with the same set of leaf weights.
Examples of Huffman Trees:
Using a Huffman Trees for Data Compression

Idea:

• Represent more common characters with shorter codes.
• Make no code the prefix of any other code, so a decoder can use the Huffman tree to read bits one at a time and always know where one code ends and the next starts.

For example, suppose that a string is composed from an alphabet of 5 characters with frequencies as shown on the left in the figure below (half the characters are A's, 1/16 of the characters are B's, etc.). Then binary codes can be assigned that are represented by the Huffman trie shown on the right.

<table>
<thead>
<tr>
<th>Character</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1/2</td>
</tr>
<tr>
<td>B</td>
<td>1/16</td>
</tr>
<tr>
<td>C</td>
<td>1/16</td>
</tr>
<tr>
<td>D</td>
<td>1/8</td>
</tr>
<tr>
<td>E</td>
<td>1/8</td>
</tr>
<tr>
<td>F</td>
<td>1/8</td>
</tr>
</tbody>
</table>

Huffman Tree:

```
          1
         /|
        / | 1
       /  |
      0   0
     /|
    /  |
   0   1
  /|
 / |
A B C D E F
```
Huffman Tree Construction Algorithm

It can be proved that the following "bottom-up" greedy algorithm produces a Huffman tree:

Initialize a FOREST of single-vertex binary trees, one for each weight.

while the FOREST has more than one tree do begin

Let $X$ and $Y$ be trees in the FOREST of lowest weight (ties broken arbitrarily).

Create a new root $r$ and make the roots of $X$ and $Y$ the children of $r$.

Set the weight of $r$ to be the sum of the weights of the roots of $X$ and $Y$.

end