Homework Due Oct 26

3 6 1 3 5 7 1

I get $T'$

II get Level Sequence

III Reconstruct Alphabetic Tree
A node is a □ or a ○

Two nodes are compatible if there is no □ between them

A weight sequence is a sequence of nodes with weight

Weight of a father is the sum of the two sons’ weights

Position of a father is the position of the left son

Given two adjacent nodes of equal weights, the left is considered less
Step I: Combination

Combine the min weight compatible pair until there is a tree $T'$
Step II: Get levels of $\square$ in $T'$
Step III: Reconstruction
Read the levels from left to right. Use a stack. 2, 3, 4, 4, 2, 2
Step I: Combination

Instead of min weight compatible pair, use a local min weight compatible pair in a weight sequence.

\[ a, b, c, d \]

If \( a > c \) and \( b < d \), then \( b \) and \( c \) is a local minimum.
Cost of a tree from a set $\leq$ cost of a tree from a sequence.
Two circle at the bottom
One circle and one square
Two squares on the slope
For an increasing weight sequence, Huffman = Hu-Tucker, so the tree $T'$ is optimum.

For a valley sequence, the tree $T'$ is optimum.
$T'$ can be normalized.

$T'$:

```
  13
 /   \
/     \
8-------5
 |       |
4 2 3 4
```

$T_n$:

```
  13
 /   \
/     \
6-------7
 |       |
4 2 3 4
```