Announcements

- No office hours today.
  *(Tomorrow?)*

- Checkpoint today.

- Program due Sunday.

- Redo on 1 problem from midterm — due next Monday.

AVL trees

Main property:
How to mess up balancing?
So: consider the lowest node which does not satisfy height-balance property \( \overline{z} \). Let \( y \) be \( z \)'s child with larger height.

Let \( x \) be \( y \)'s child with larger height. Now – fix it!
What did you do?

- 44
  - 17
    - 32
  - 78
    - 50
      - 40
      - 62
Generalize - Consider x, y, a, 2. How can we restructure?

(Hint: What is inorder traversal of these in each case?)
Actual picture:

Where do the sub trees go??
Any way you do this, "2" becomes the root of the new subtree with "1" to the left and "3" to the right! What about T1, T2, T3, and T4?
So how can we code this?

Back to Binary Tree.h:
- pivot (it)
- and its parent

it

11

12

13
Removing in AVL trees

Step 1: Remove - just like in BST

Step 2: Re-balance (if removal violated H-B property.)

Note: Unlike insert, remove could actually un-balance all the way to the root.
Example: remove (44)

Then: remove (17)
Fixing the tree