Finishing last lecture
Height of a Level-Balanced Tree

$n_L$: fewest nodes to have root at level L

$n_1$: 

$n_2$: 

$n_3$: 3

$n_4$: is smallest w/ root at L=2

$L > 4$: 

\[ n_L = 1 + 2n_{L-2} \]
What is a priority queue?

- `insert(e)`: inserts given element into PQ.
- `min()`: returns reference to a smallest elt
- `extractMin()`: remove what `min()` returns.
What is a complete binary tree?

What is the min heap property?
Heap as std::vector

- Where is root?
- Where is i's left child? $2i$
- Where is i's right child? $2i + 1$
Implementing the Priority Queue

- Insert ‘7’
  
  *Hint: must have heap after*

- Then insert ‘4’ ‘2’ ‘6’
Implementing the Priority Queue

Find min in a heap
Implementing the Priority Queue

- extract-min from heap

*Hint: must have heap after*
XKCD # 835: Tree. Not only is that terrible in general, but you just KNOW Billy’s going to open the root present first, and then everyone will have to wait while the heap is rebuilt.
Two min-heaps, A and B
Do they have a common element?
Use only the priority queue API

```java
while A and B both not empty:
    if A.min() == B.min() return true
    if A.min() < B.min():
        A. extractMin()
    else:
        B. extractMin()
return false
```