

# **CSE 373**

**MARCH 31 – PRIORITY QUEUES AND  
THE HEAP**

# ASSORTED MINUTIAE

- **Weiss readings**
  - On course website
- **HW1 is out (patch out tonight)**
  - Eclipse users will get some help
  - Test 5 fix
- **143 review**
  - Monday, time/location TBA

# TODAY'S LECTURE

- **Priority Queue ADT**
- **Heap DS**
  - Heap Property
  - Completeness property
- **Implementation**

# REVIEW FROM LAST WEEK

- **Priority Queue**
  - Data enqueued with a priority
  - Lower priority data dequeue first
  - Maintain queue principle?
- **Implementations?**
  - Array and Linked List both have serious flaws.

# PRIORITY QUEUE

- **Priority queue implementations?**
  - Binary search tree?
    - Faster insert
    - Find? Always deleting the smallest (left-most) element
    - Maintaining FIFO?
    - Changing priority?

# PRIORITY QUEUE

- **Want the speed of trees (but not BST)**
- **Priority Queue has unique demands**
- **Other types of trees?**
- **Review BST first**

# PROPERTIES (BST)

- **Tree (Binary)**
  - Root
  - (Two) Children
  - No cycles
- **Search**
  - Comparable data
  - Left child data  $<$  parent data
  - Smallest child is at the left most node

# PROPERTIES (BST)

- **Binary tree may be useful**
- **Search property doesn't help**
  - Always deleting min
  - Put min on top!



# HEAP-ORDER PROPERTY

- **Still a binary tree**
- **Instead of search (left < parent),  
parent should be less than children**
- **How to implement?**
- **Insert and delete are different than BST**

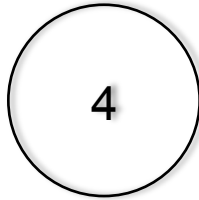
# HEAP-ORDER PROPERTY

- **Still a binary tree**
- **Instead of search (left < parent),  
parent should be less than children**
- **How to implement?**
- **Insert and delete are different than BST**

# HEAP EXAMPLE

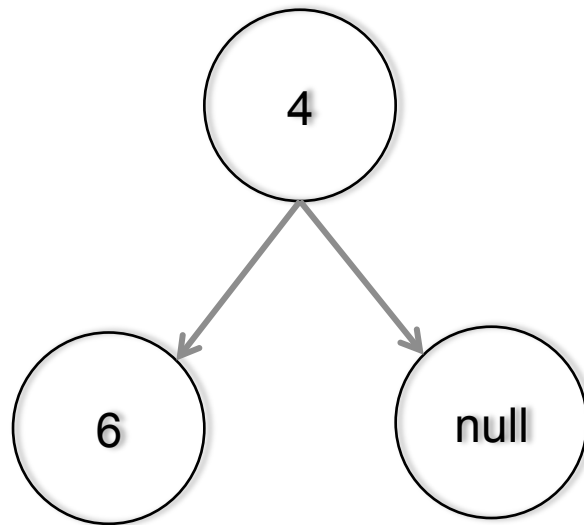
- Only looking at priorities
- Insert something priority 4

# HEAP EXAMPLE



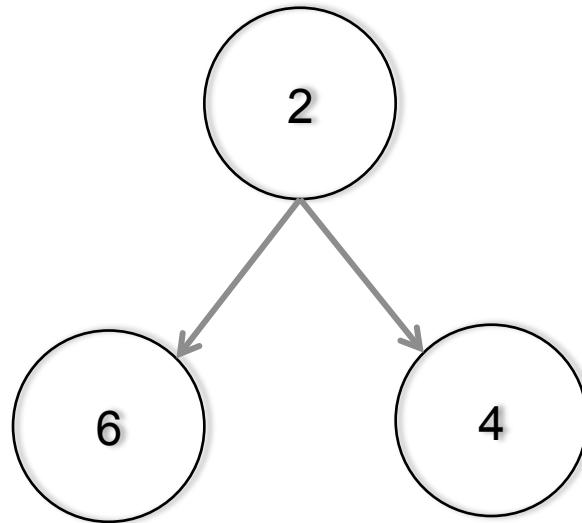
- **Now insert priority 6?**
- **Should come after 4, but no preference right over left?**
- **Solution: fill the tree from top to bottom left to right.**

# HEAP EXAMPLE



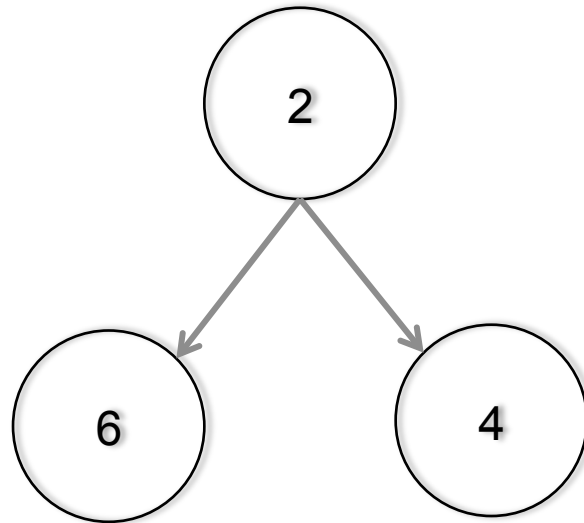
**Now insert 2.**

# HEAP EXAMPLE



**Now insert 2.**

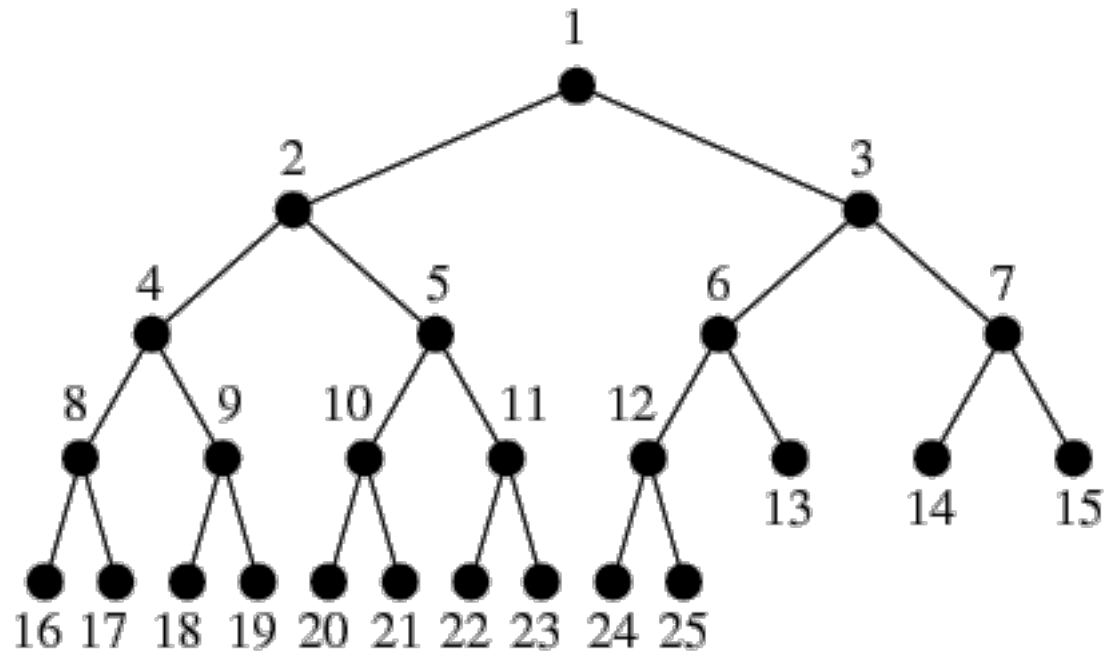
# HEAP EXAMPLE



**Is this the only solution that maintains the heap property?**

**Is any one better than the other?**

# COMPLETENESS



**Filling left to right and top to bottom is another property - completeness**

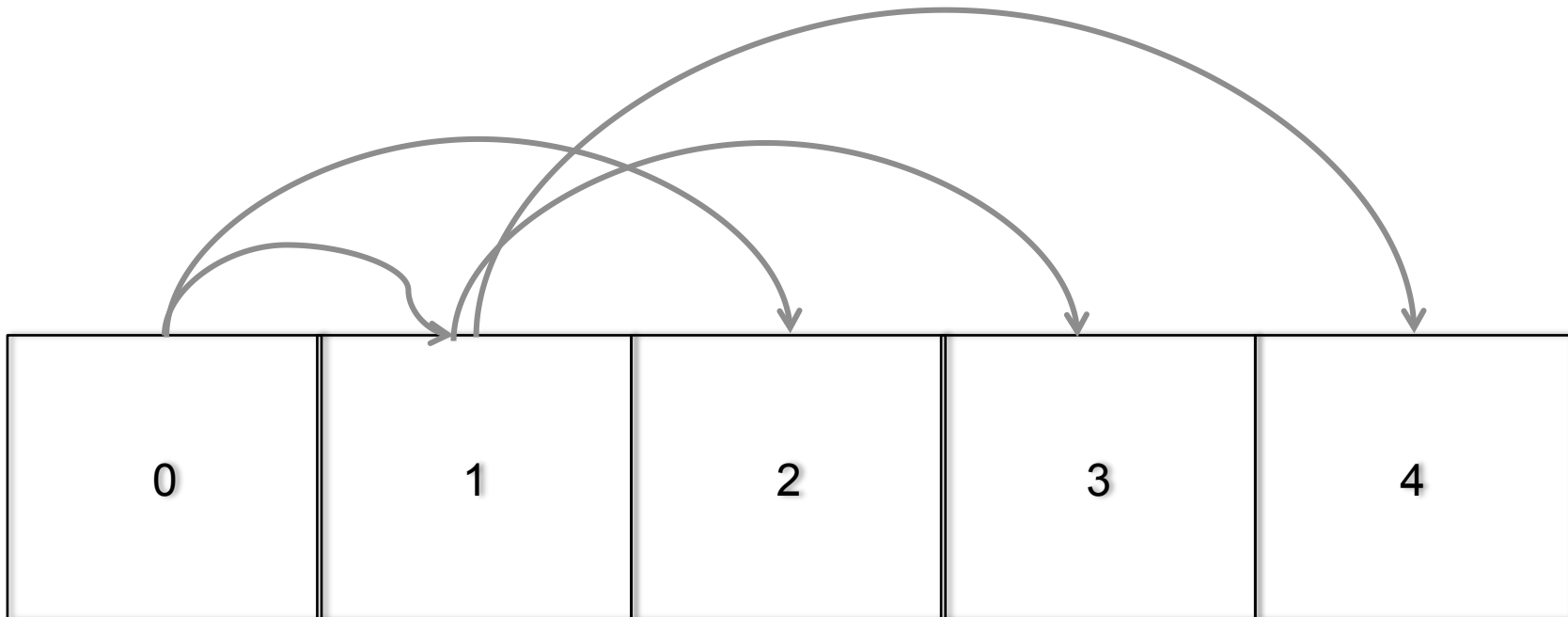


# HEAPS

- **Heap property (parents  $<$  children)**
- **Complete tree property (left to right, bottom to top)**
- **How does this help?**
  - Array implementation

# HEAPS

- Insert into array from left to right
- For any parent at index  $i$ , children at  $2*i+1$  and  $2*i+2$



# HEAPS

- **How to maintain heap property then?**
  - Parent must be higher priority than children
- **Two functions – percolate up and percolate down**
  - DRAWN NOTES HERE

# HEAPS

- **Does the heap work for the Priority Queue problem?**
  - FIFO preservation?

**No. Only comparisons are priority.**

# **NEXT WEEK**

- **Look more closely at heap functions and runtimes**
- **Beginning of algorithm analysis**