

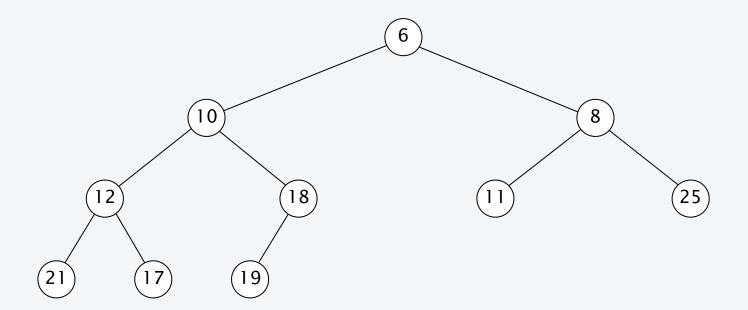
## **DATA STRUCTURES II**

binary heap demo

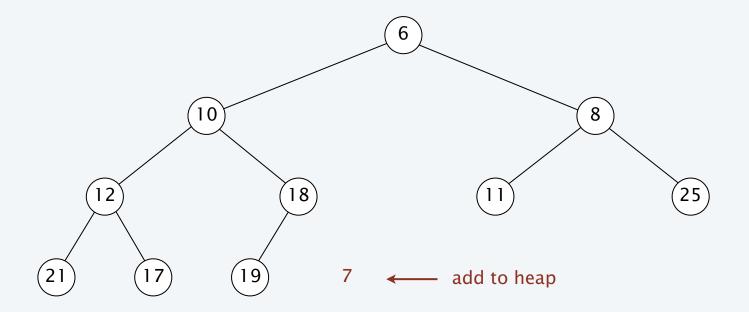
Copyright © 2013 Kevin Wayne

http://www.cs.princeton.edu/~wayne/kleinberg-tardos

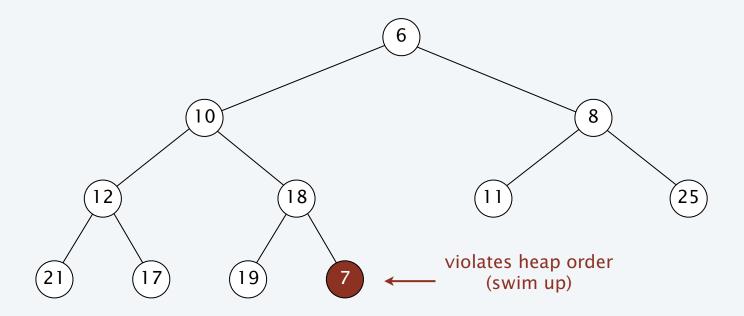
### heap ordered



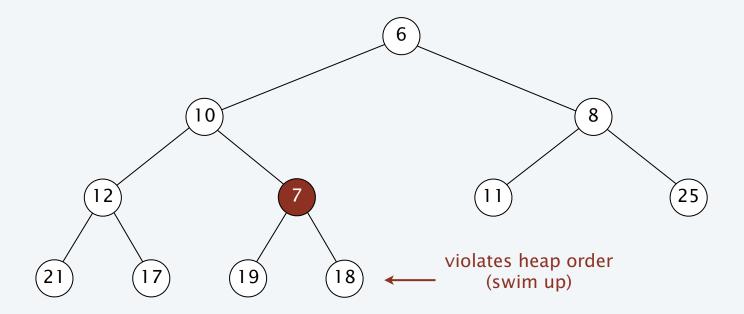
Insert. Add node at end; repeatedly exchange element in child with element in parent until heap order is restored.



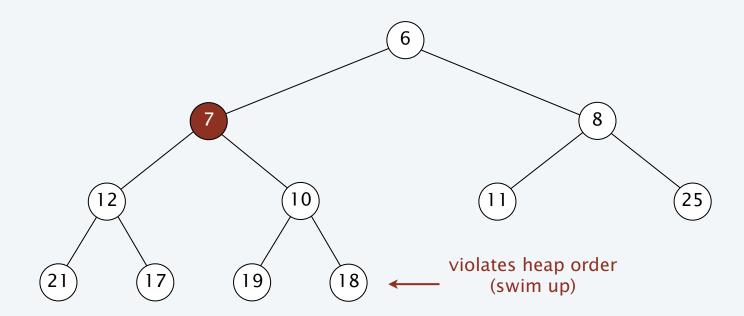
Insert. Add node at end; repeatedly exchange element in child with element in parent until heap order is restored.



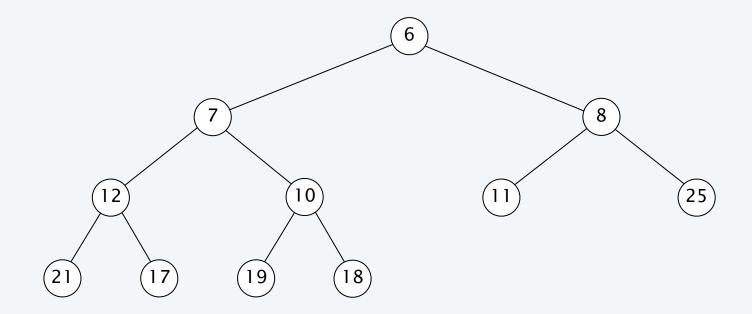
Insert. Add node at end; repeatedly exchange element in child with element in parent until heap order is restored.



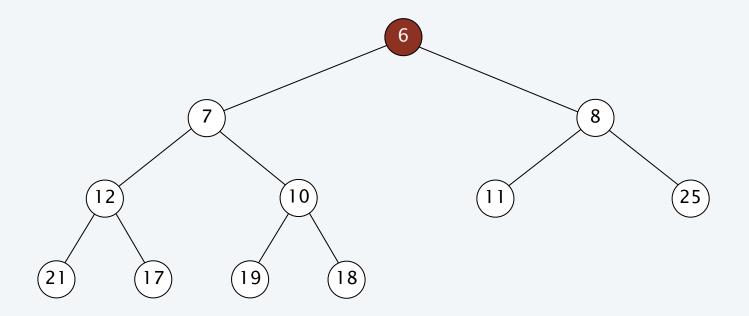
Insert. Add node at end; repeatedly exchange element in child with element in parent until heap order is restored.



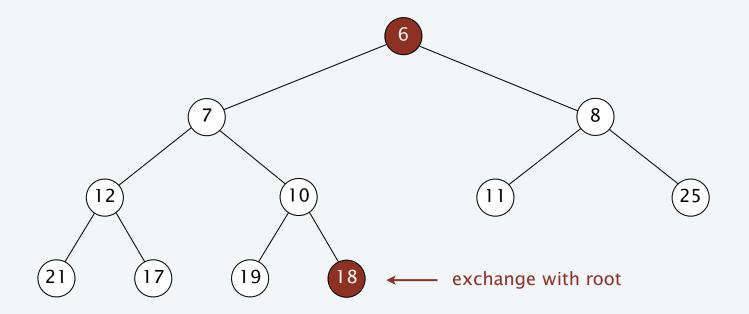
### heap ordered



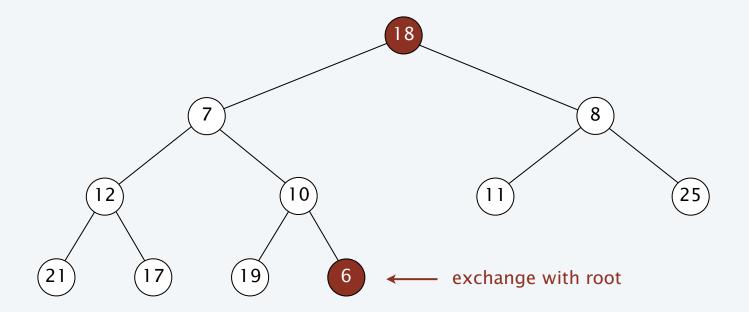
Extract min. Exchange root node with last node; repeatedly exchange element in parent with element in larger child until heap order is restored.



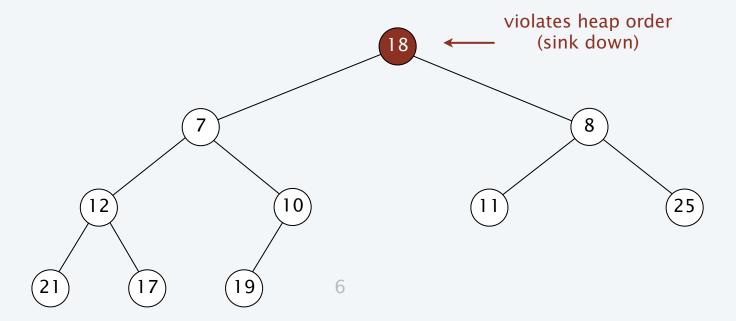
Extract min. Exchange root node with last node; repeatedly exchange element in parent with element in larger child until heap order is restored.



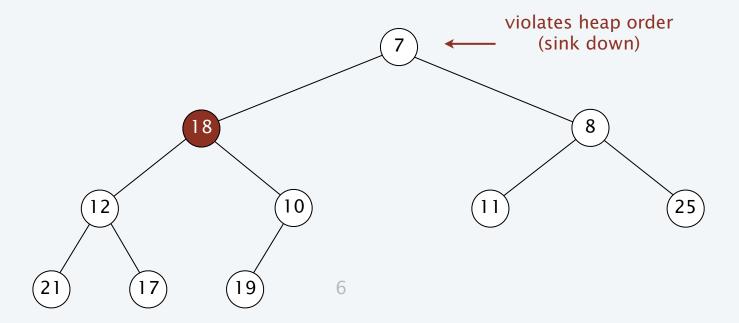
Extract min. Exchange root node with last node; repeatedly exchange element in parent with element in larger child until heap order is restored.



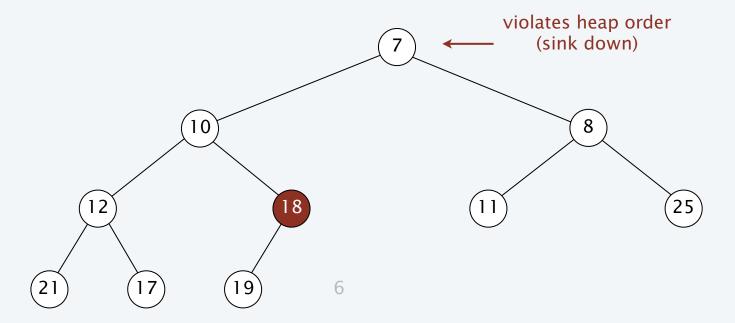
Extract min. Exchange root node with last node; repeatedly exchange element in parent with element in larger child until heap order is restored.



Extract min. Exchange root node with last node; repeatedly exchange element in parent with element in larger child until heap order is restored.



Extract min. Exchange root node with last node; repeatedly exchange element in parent with element in larger child until heap order is restored.



### heap ordered

