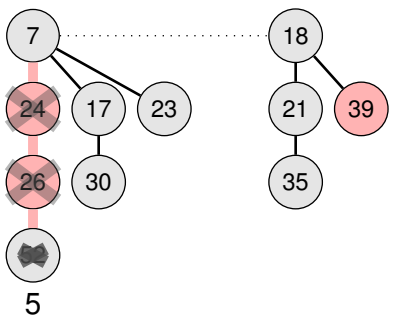
# Microchallenge 5

- Implement the Fibonacci Heap priority queue
- Implement a procedure, which for any given $k \geq 5$, performs a sequence of operations which includes a CASCADING CUT of length exactly $k$ (that is, in addition to the node whose key is being decreased, another $k-1$ (marked) ancestors have to be cut from the tree.) Your procedure should start with an empty Fibonacci Heap.
- Submit your program and include the full sequence of operations for the case $k = 6$. Fancy graphical output is not strictly required but will be appreciated (and may be uploaded to the course webpage later)
- Deadline: Submission by email before Friday February 27

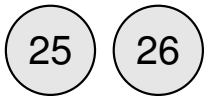


# Solution Microchallenge 5

Insert(25)

# Solution Microchallenge 5

Insert(26)

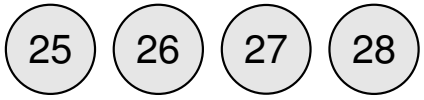


# Solution Microchallenge 5

Insert(27)

# Solution Microchallenge 5

Insert(28)

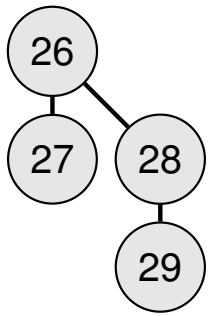


# Solution Microchallenge 5

Insert(29)

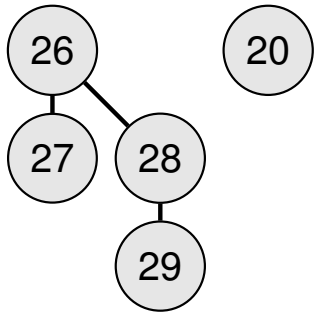
# Solution Microchallenge 5

Extract-Min



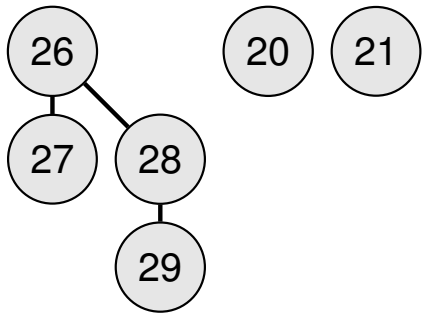
# Solution Microchallenge 5

Insert(20)



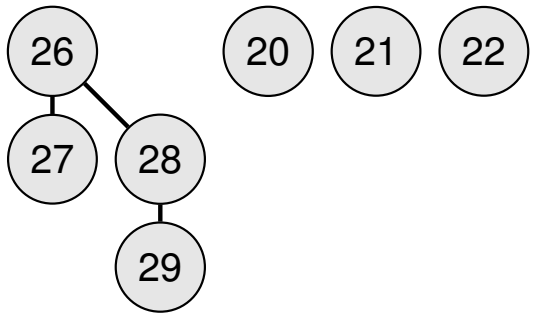
# Solution Microchallenge 5

Insert(21)



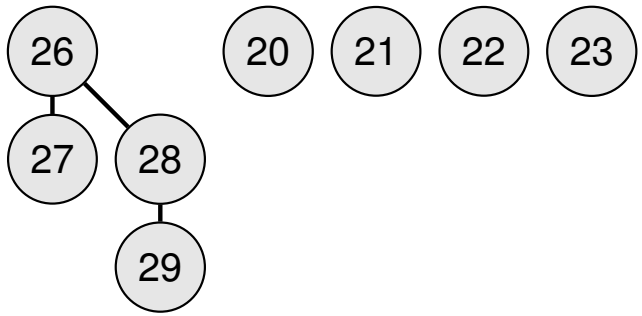
# Solution Microchallenge 5

Insert(22)



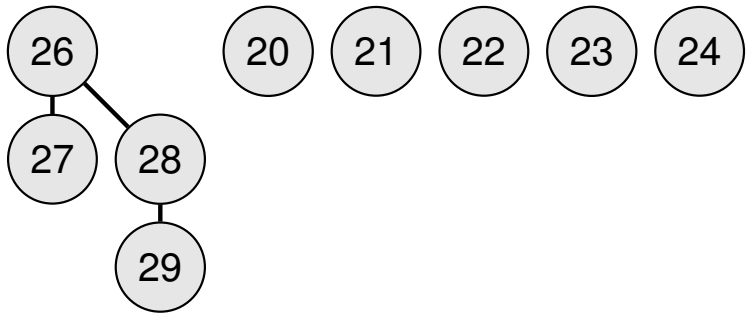
# Solution Microchallenge 5

Insert(23)



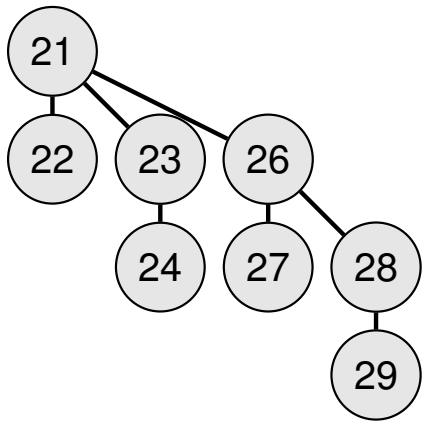
# Solution Microchallenge 5

Insert(24)



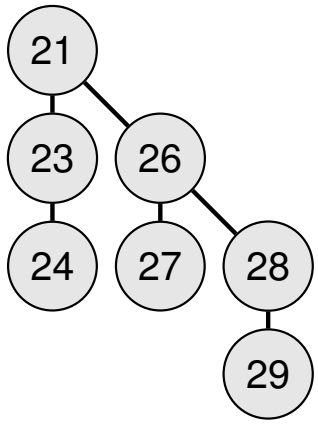
# Solution Microchallenge 5

Extract-Min



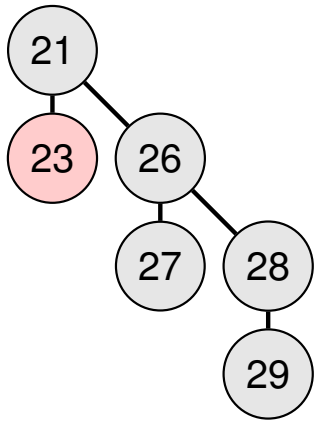
# Solution Microchallenge 5

Delete(22)



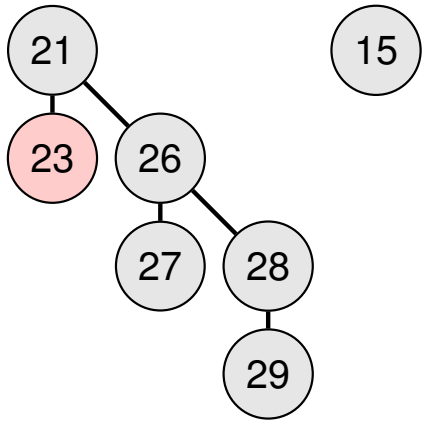
# Solution Microchallenge 5

Delete(24)



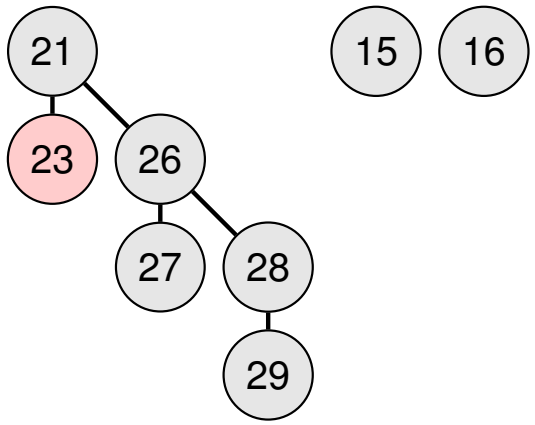
# Solution Microchallenge 5

Insert(15)



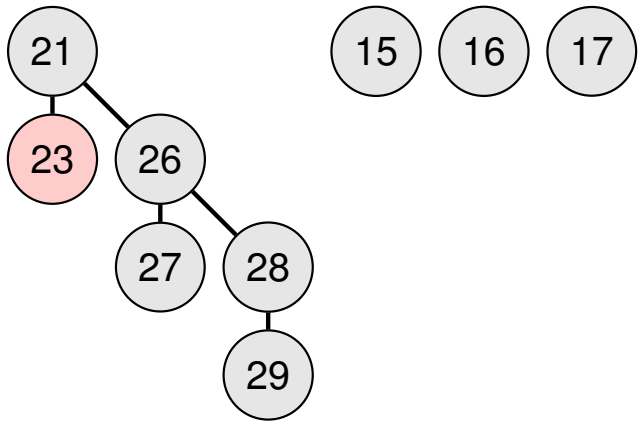
# Solution Microchallenge 5

Insert(16)



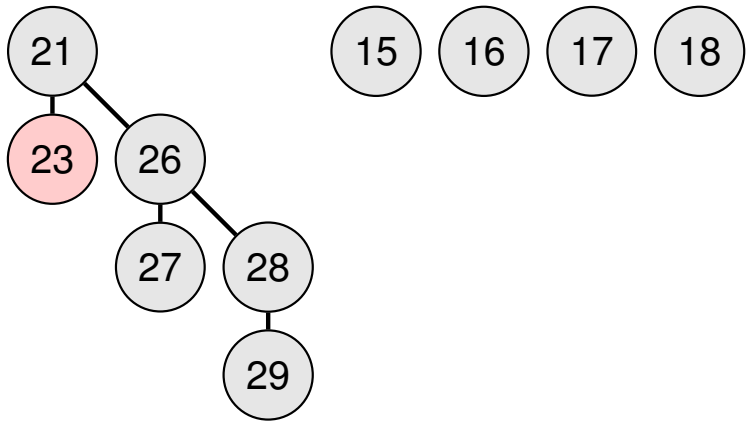
# Solution Microchallenge 5

Insert(17)



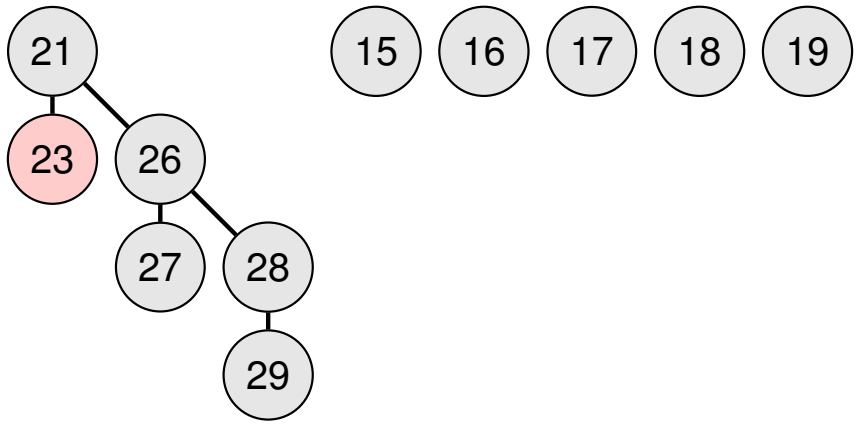
# Solution Microchallenge 5

Insert(18)



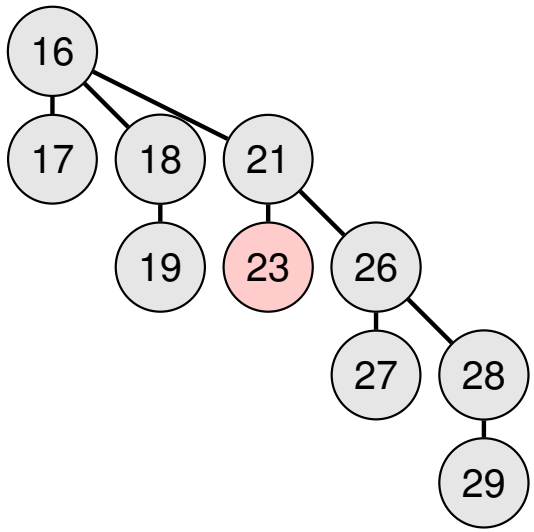
# Solution Microchallenge 5

Insert(19)



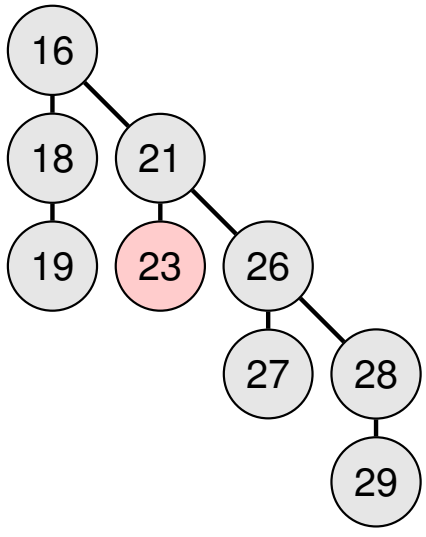
# Solution Microchallenge 5

Extract-Min



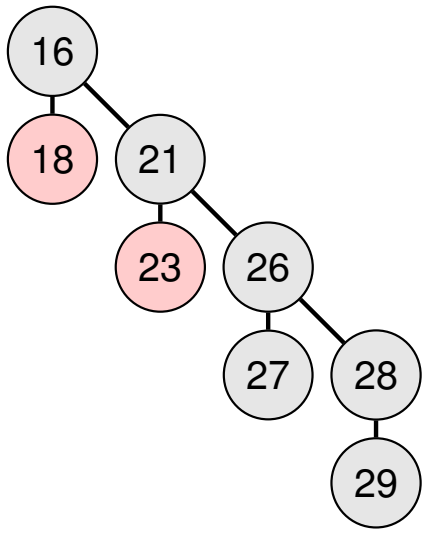
# Solution Microchallenge 5

Delete(17)



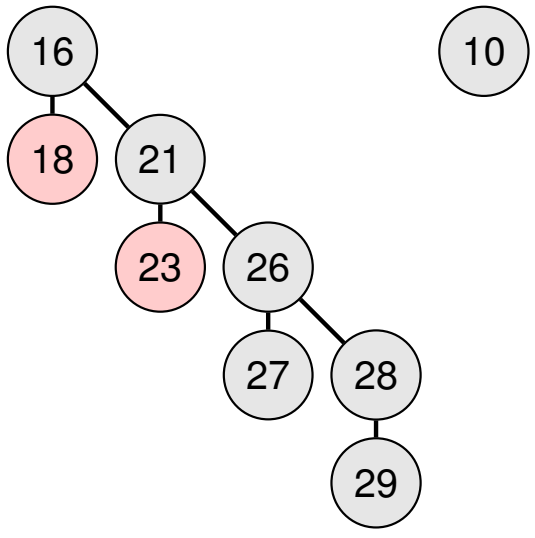
# Solution Microchallenge 5

Delete(19)



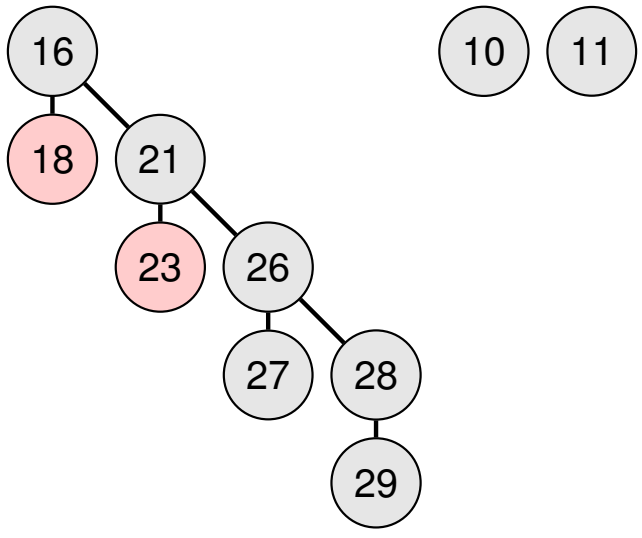
# Solution Microchallenge 5

Insert(10)



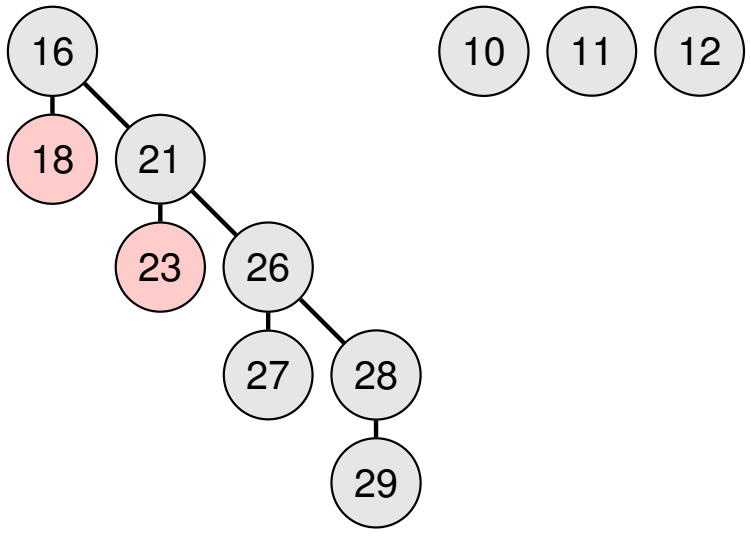
# Solution Microchallenge 5

Insert(11)



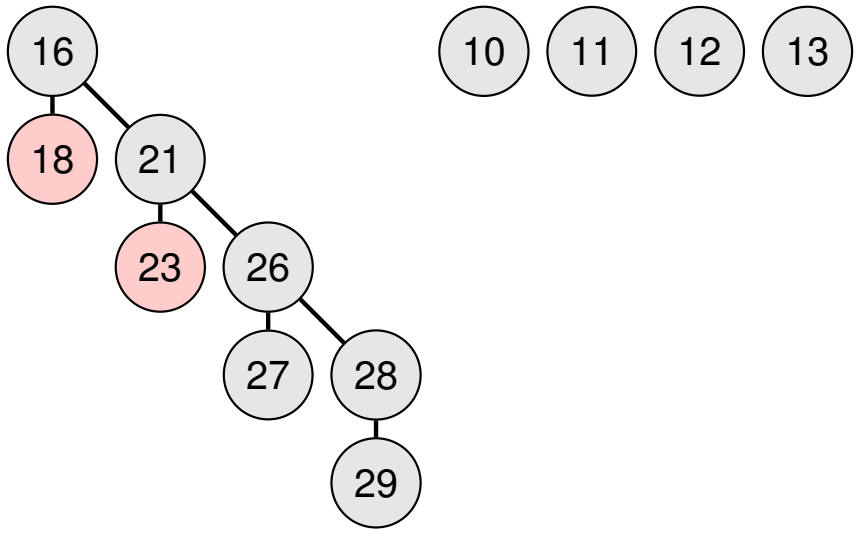
# Solution Microchallenge 5

Insert(12)



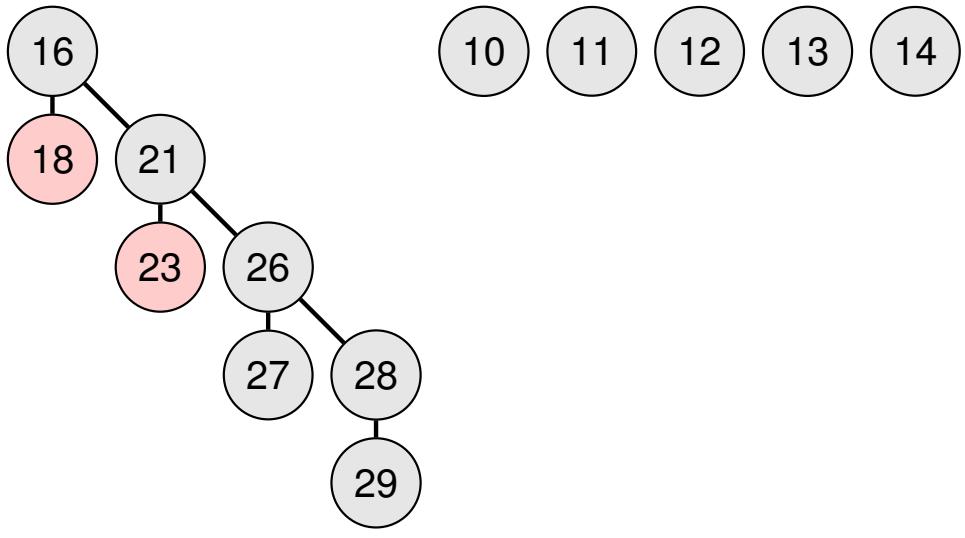
# Solution Microchallenge 5

Insert(13)



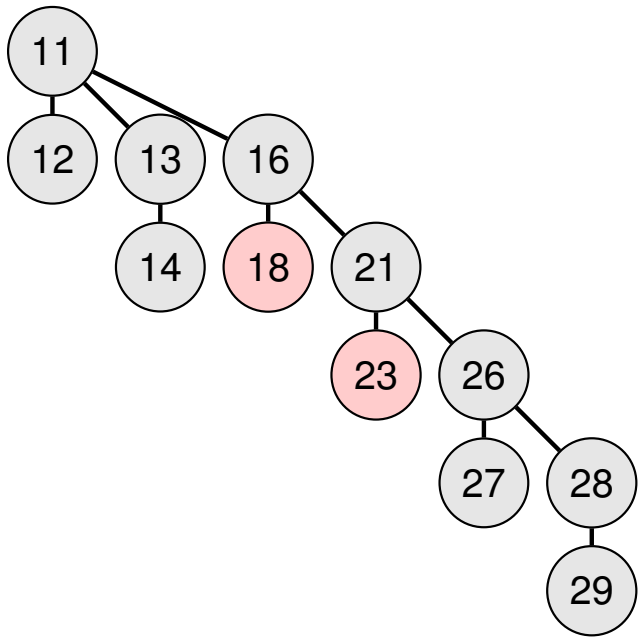
# Solution Microchallenge 5

Insert(14)



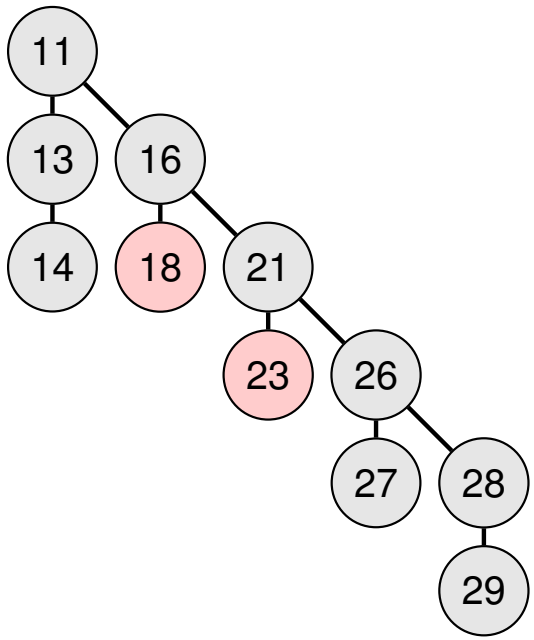
# Solution Microchallenge 5

Extract-Min



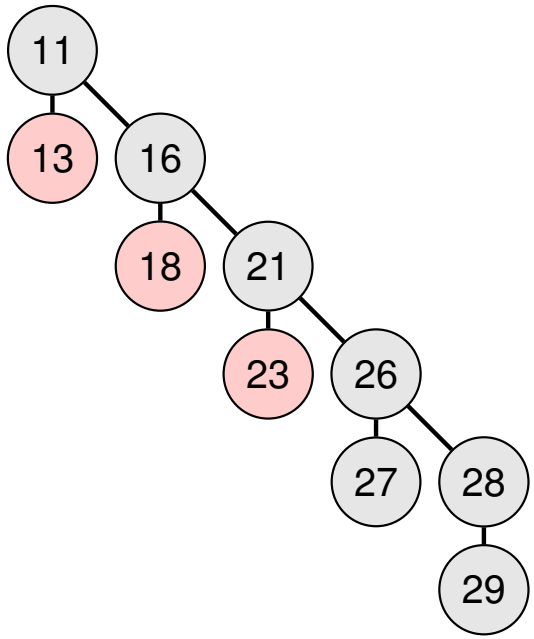
# Solution Microchallenge 5

Delete(12)



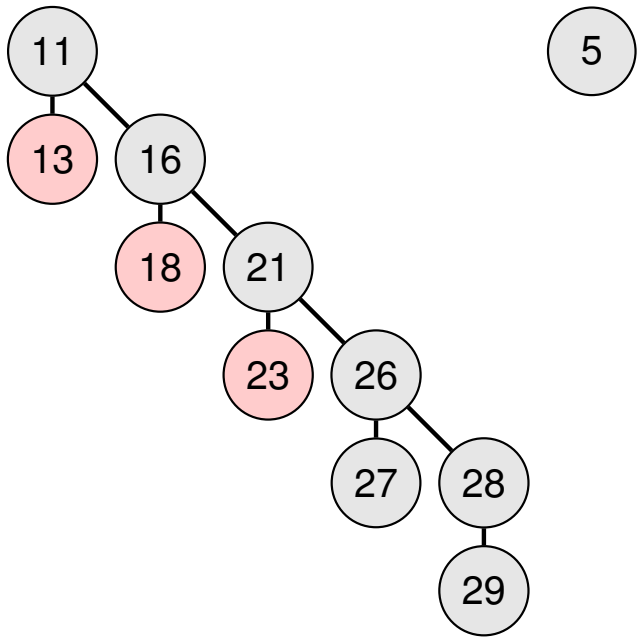
# Solution Microchallenge 5

Delete(14)



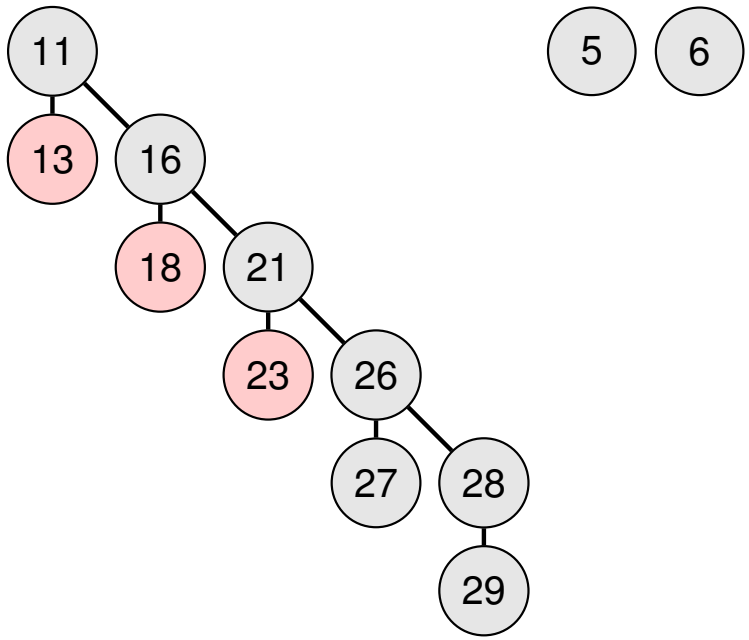
# Solution Microchallenge 5

Insert(5)



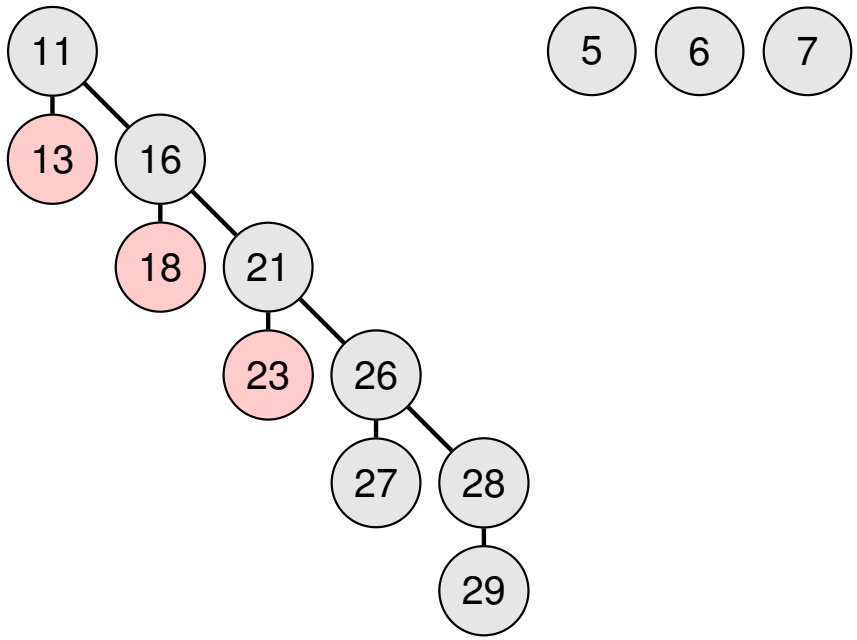
# Solution Microchallenge 5

Insert(6)



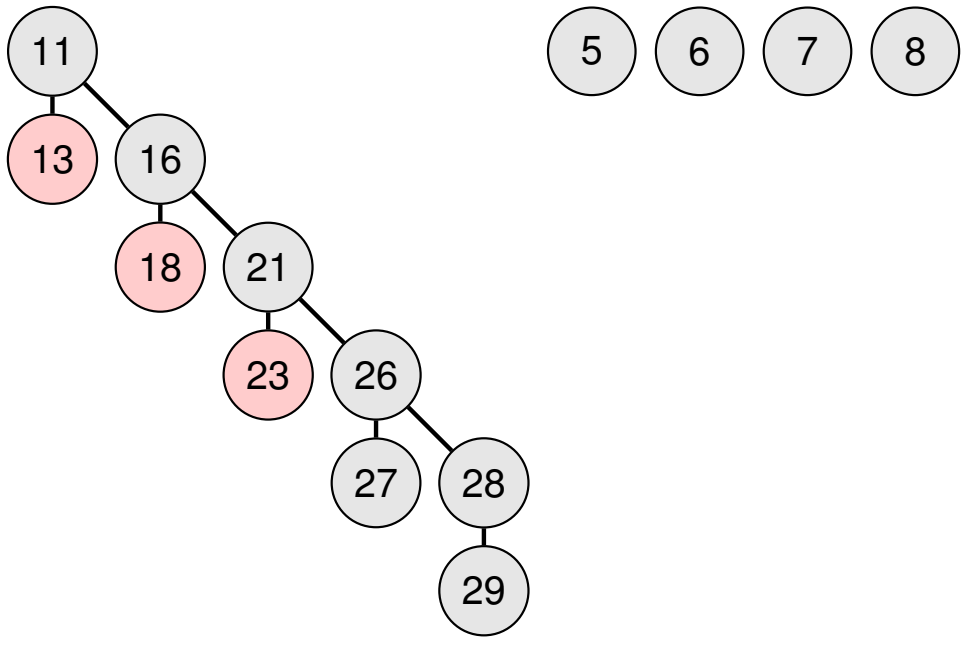
# Solution Microchallenge 5

Insert(7)



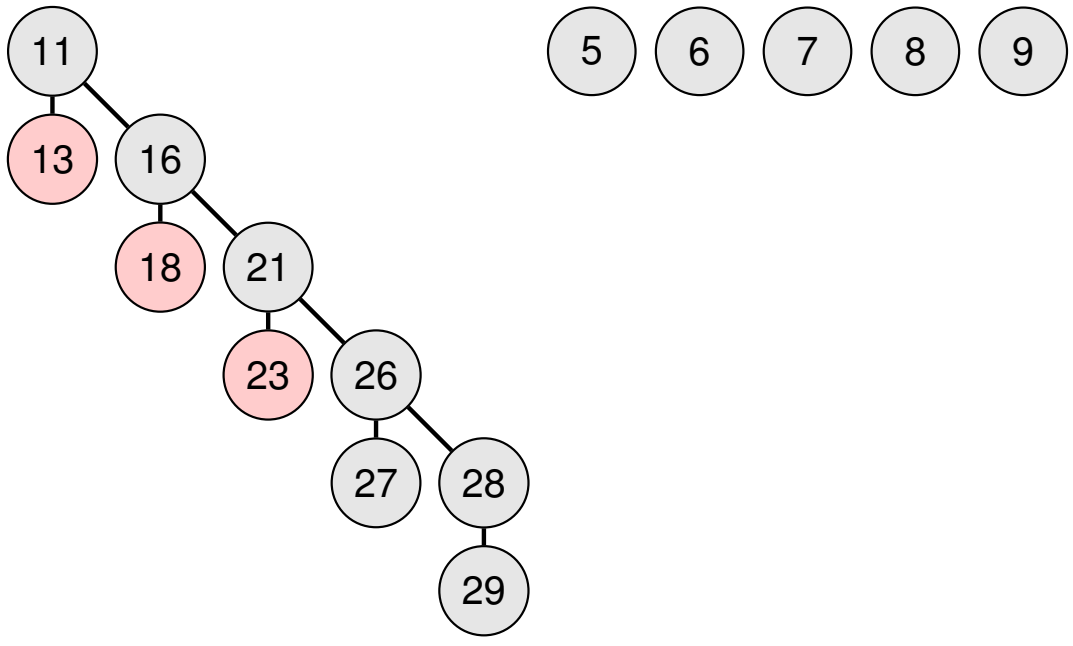
# Solution Microchallenge 5

Insert(8)



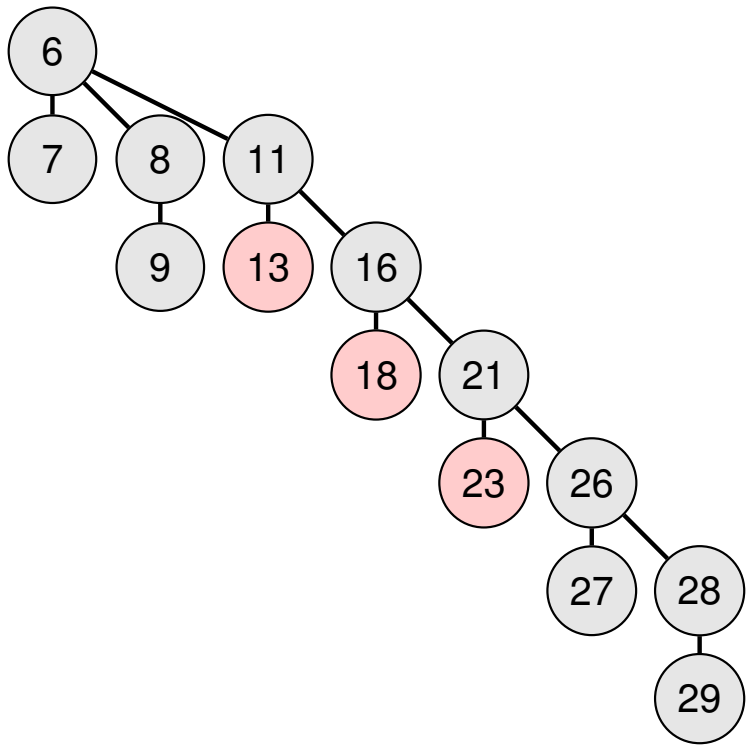
# Solution Microchallenge 5

Insert(9)



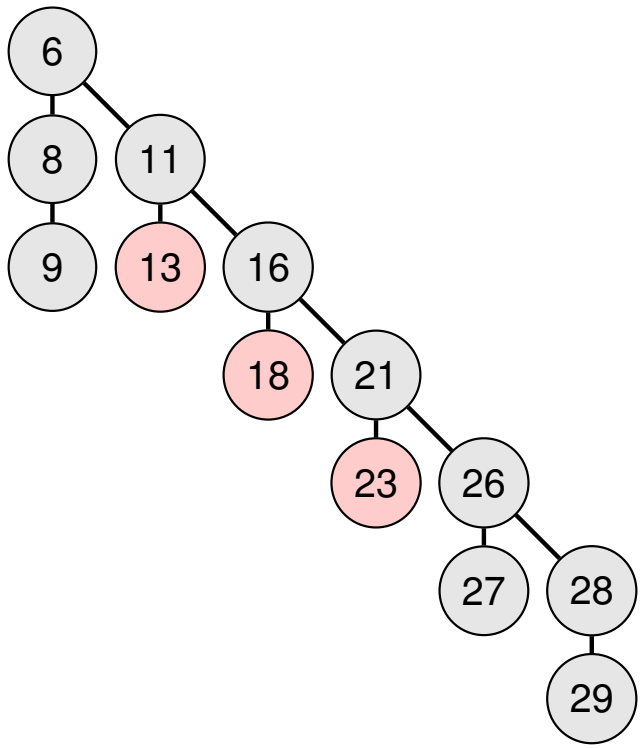
# Solution Microchallenge 5

Extract-Min



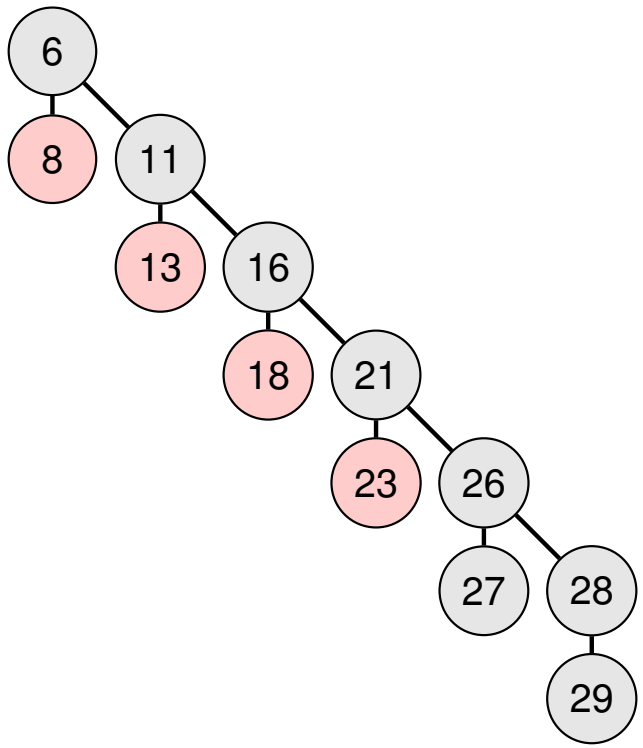
# Solution Microchallenge 5

Delete(7)



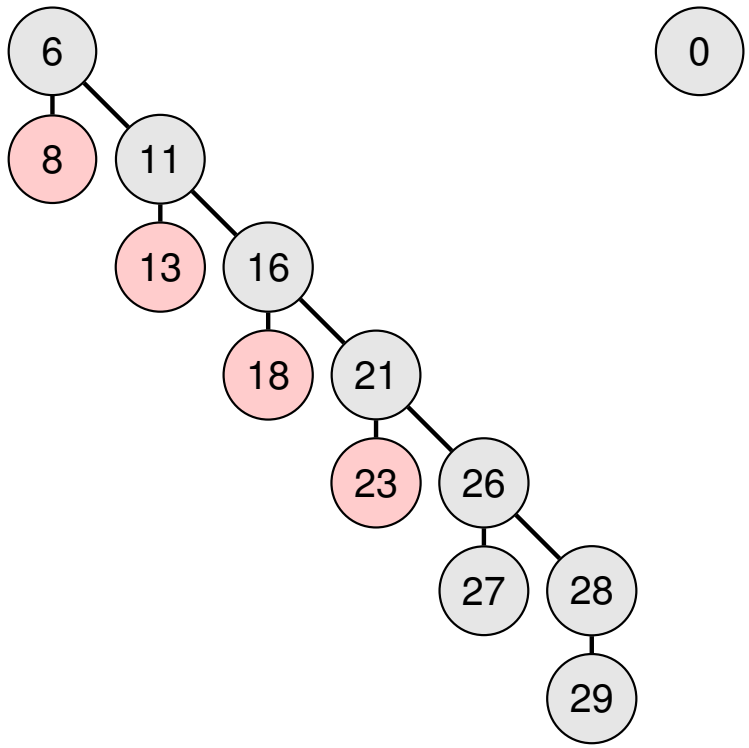
# Solution Microchallenge 5

Delete(9)



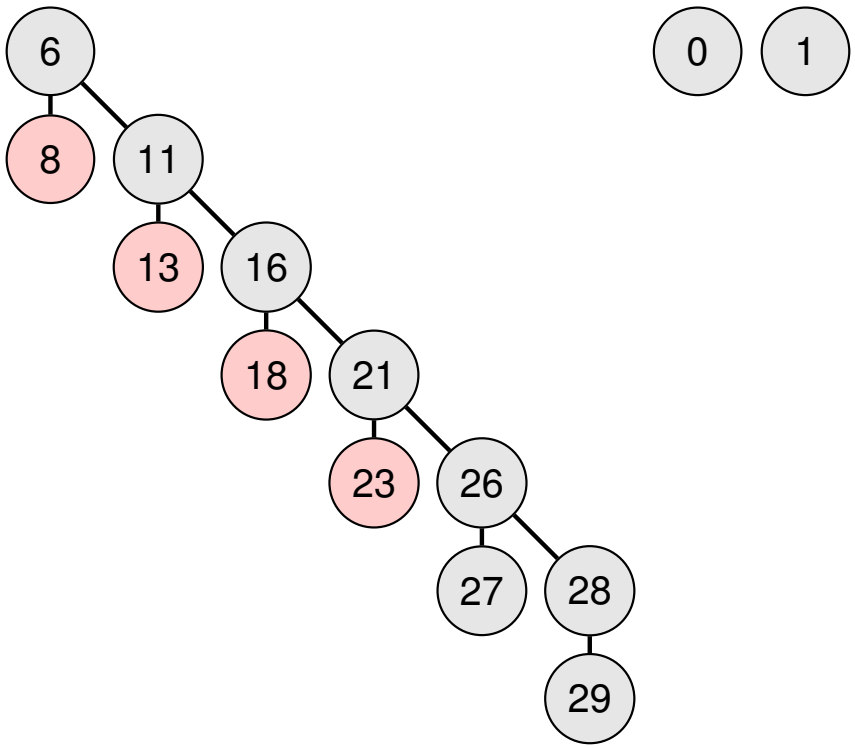
# Solution Microchallenge 5

Insert(0)



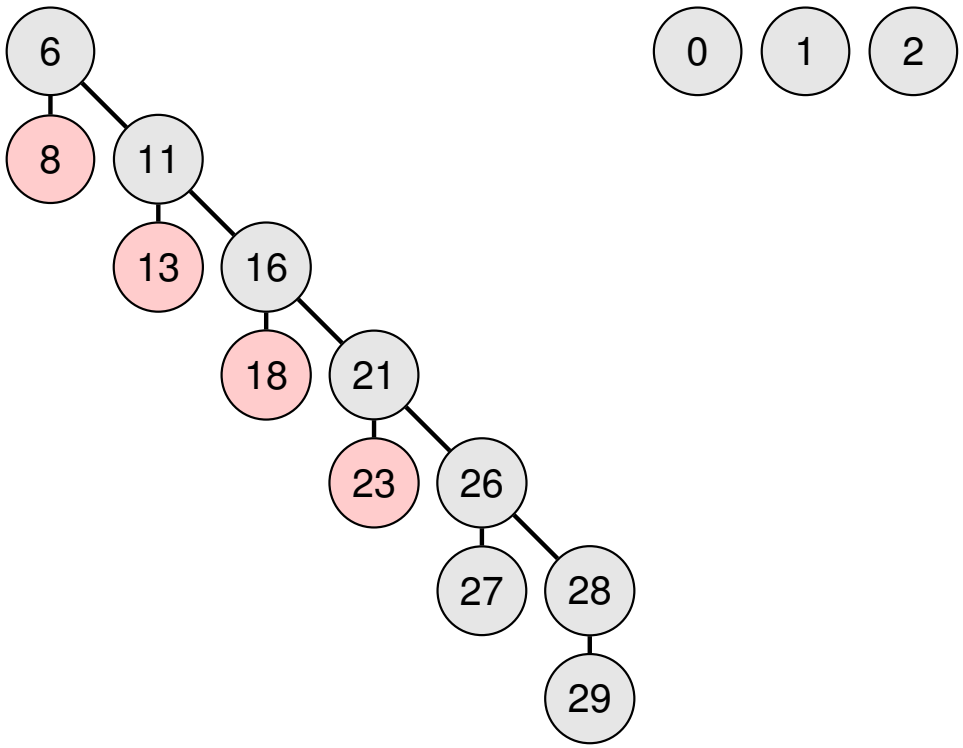
# Solution Microchallenge 5

Insert(1)



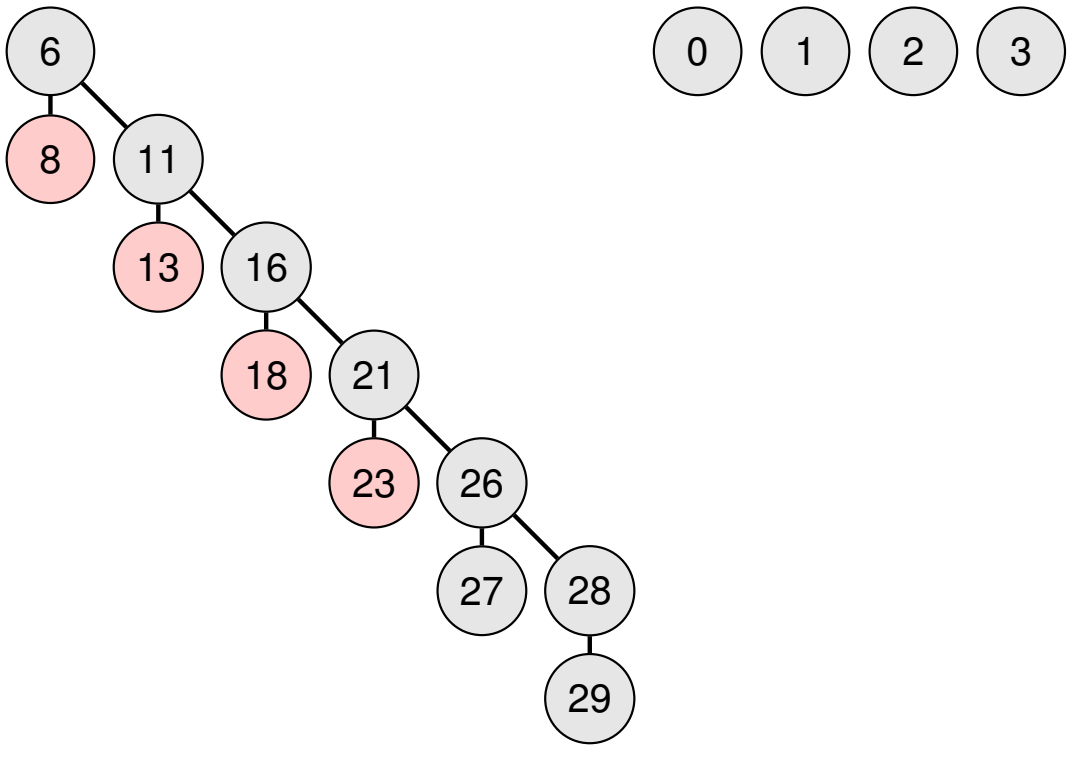
# Solution Microchallenge 5

Insert(2)



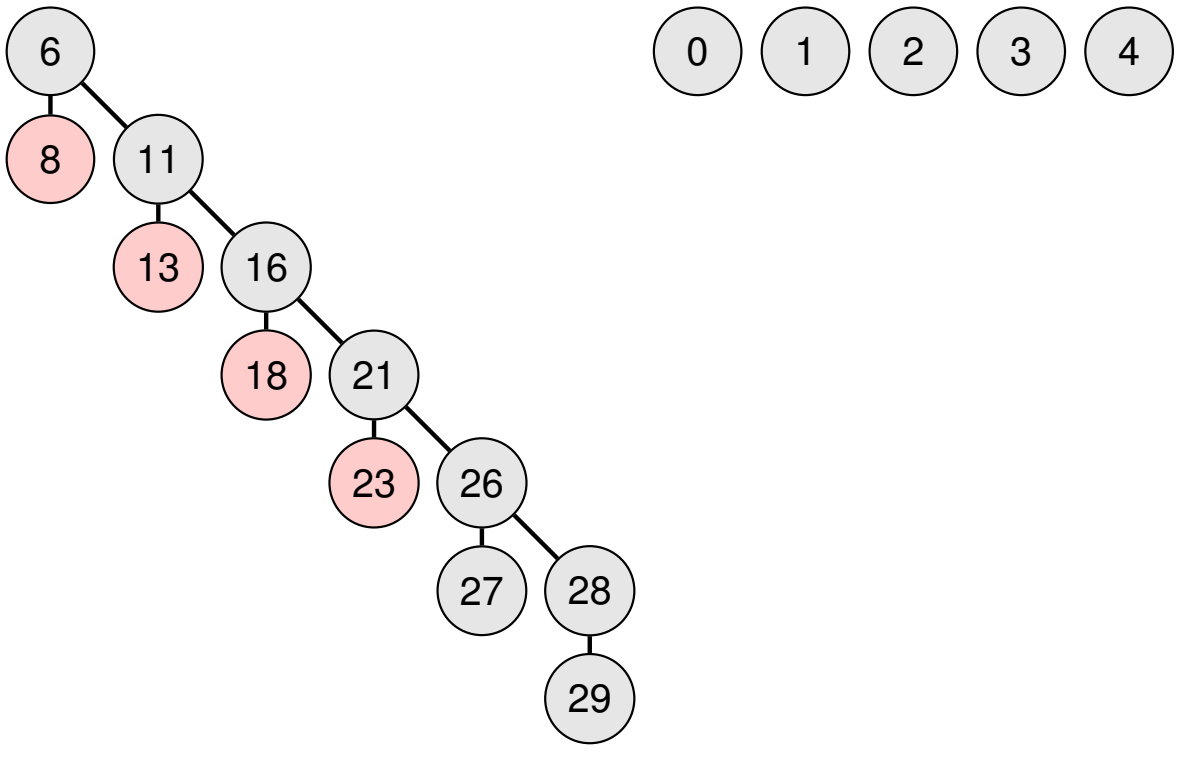
# Solution Microchallenge 5

Insert(3)



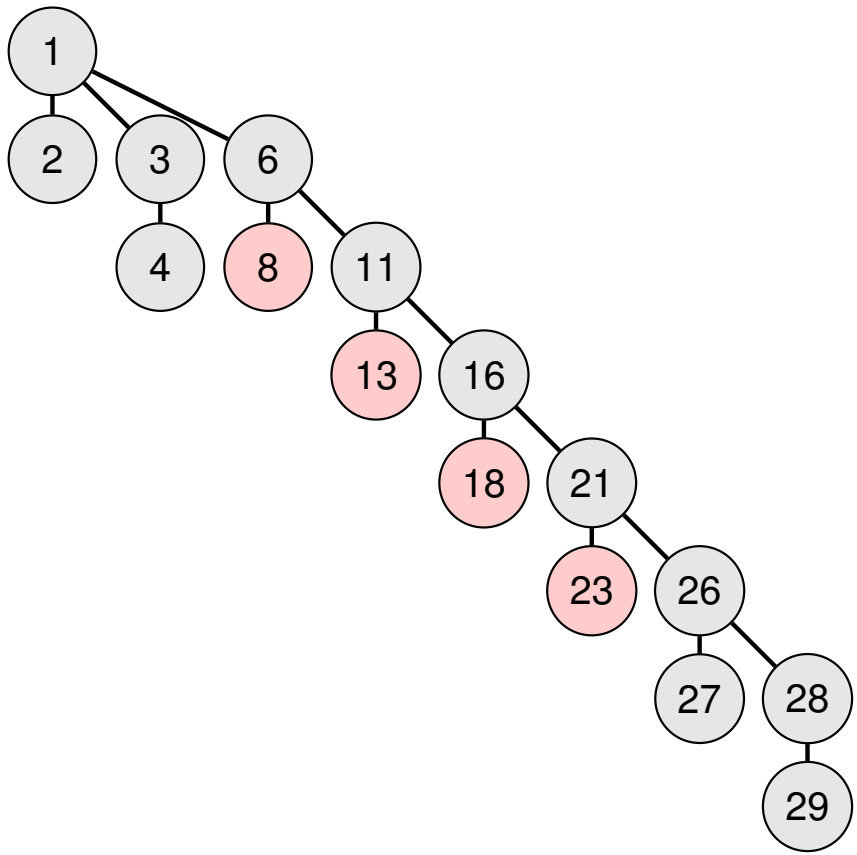
# Solution Microchallenge 5

Insert(4)



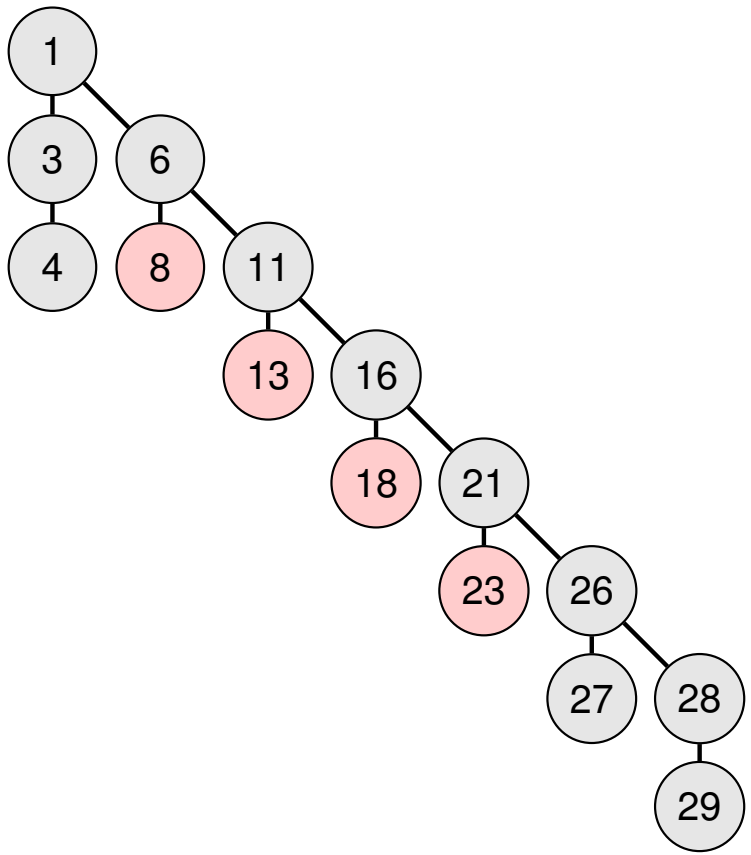
# Solution Microchallenge 5

Extract-Min



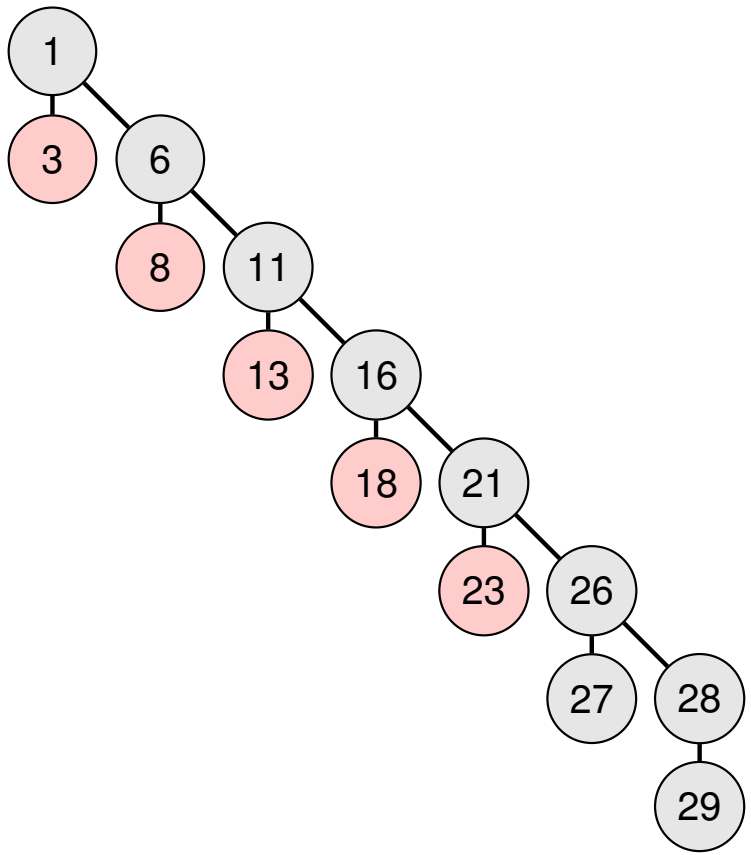
# Solution Microchallenge 5

Delete(2)



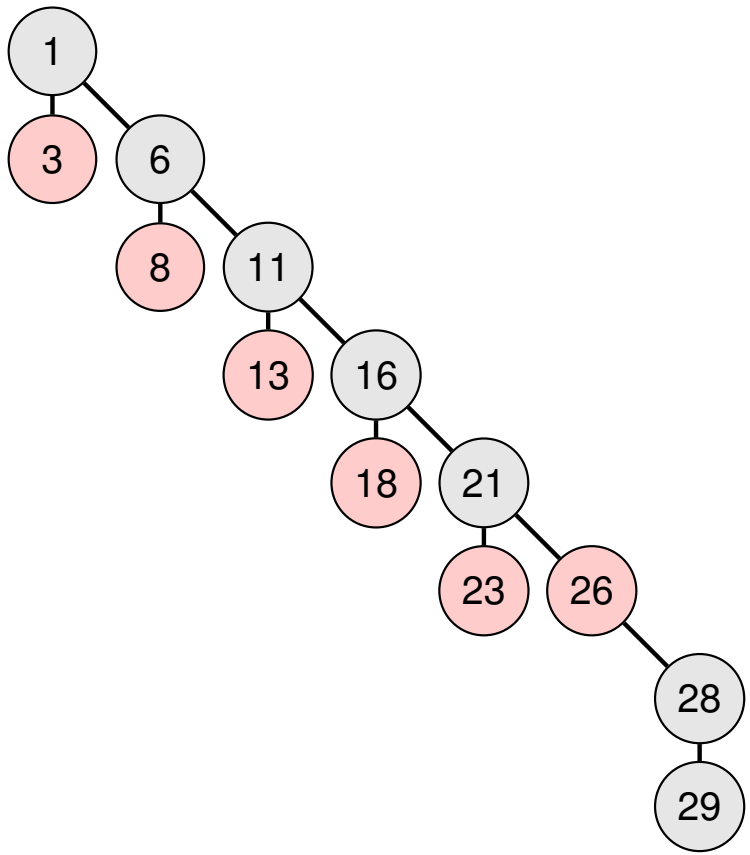
# Solution Microchallenge 5

Delete(4)



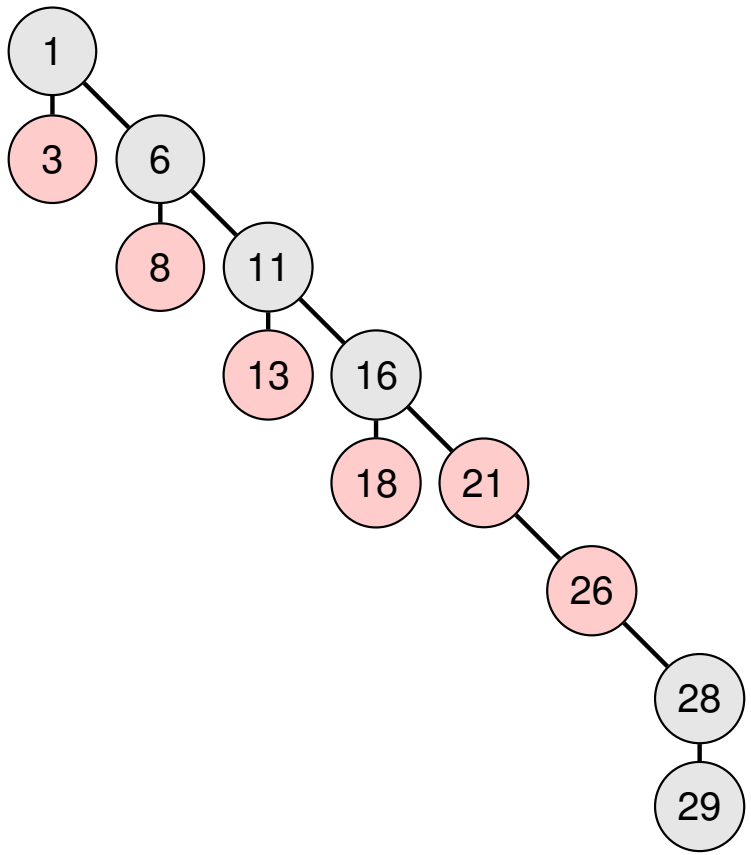
# Solution Microchallenge 5

Delete(27)



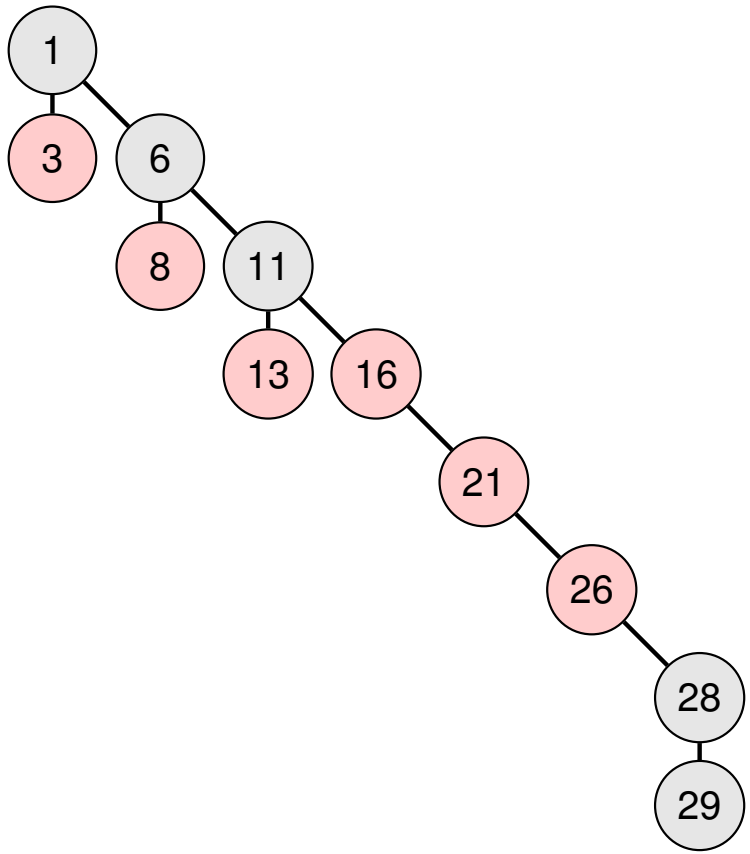
# Solution Microchallenge 5

Delete(23)



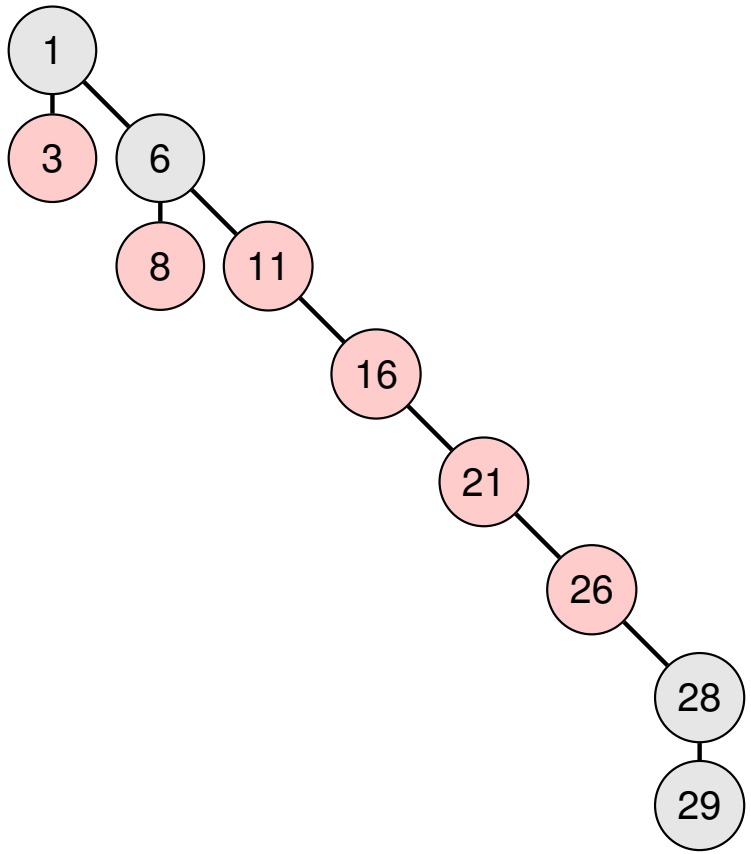
# Solution Microchallenge 5

Delete(18)



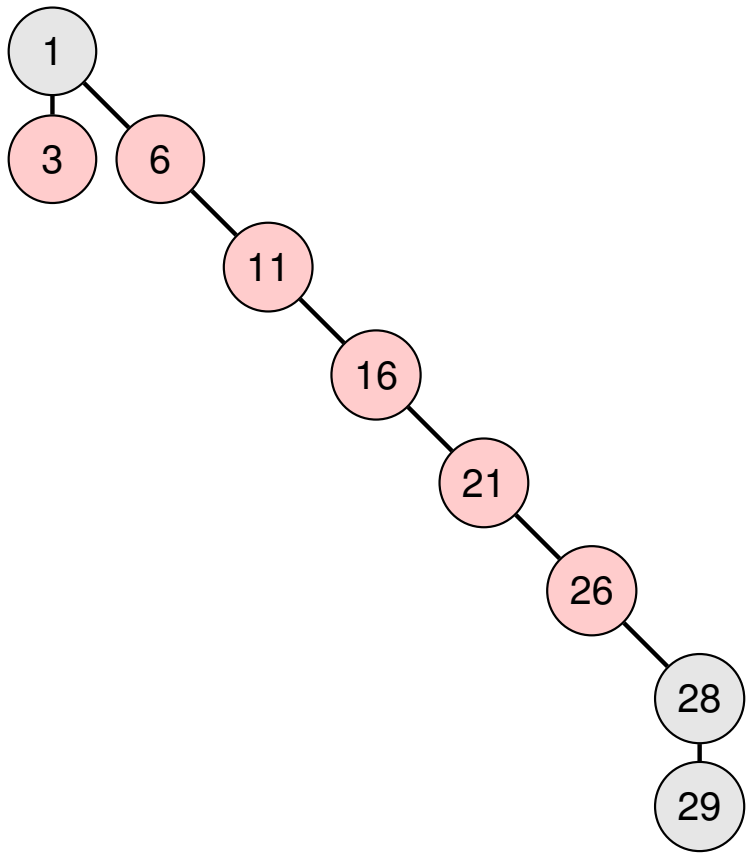
# Solution Microchallenge 5

Delete(13)



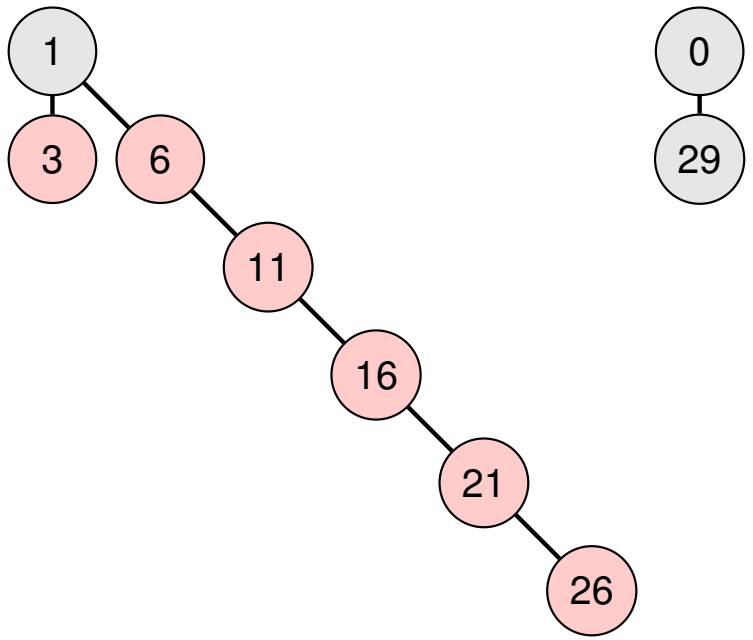
# Solution Microchallenge 5

Delete(8)



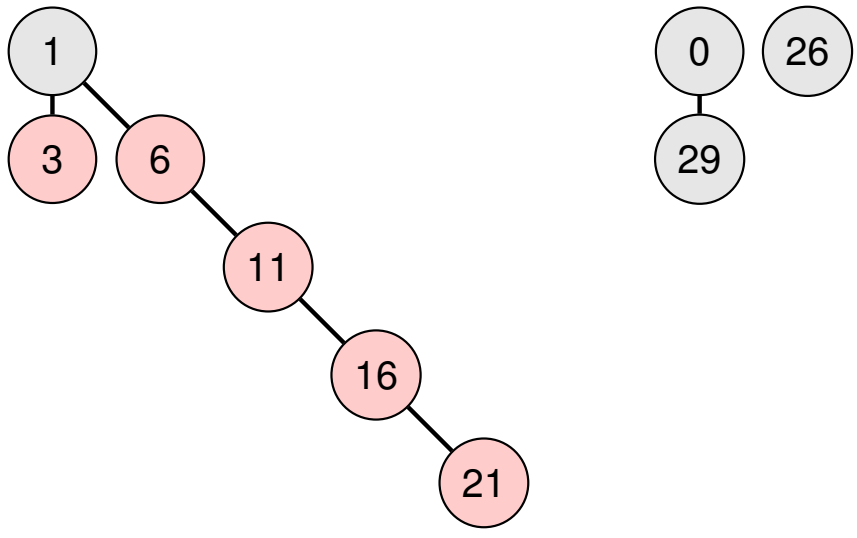
# Solution Microchallenge 5

Decrease-Key(28 $\rightarrow$ 0)



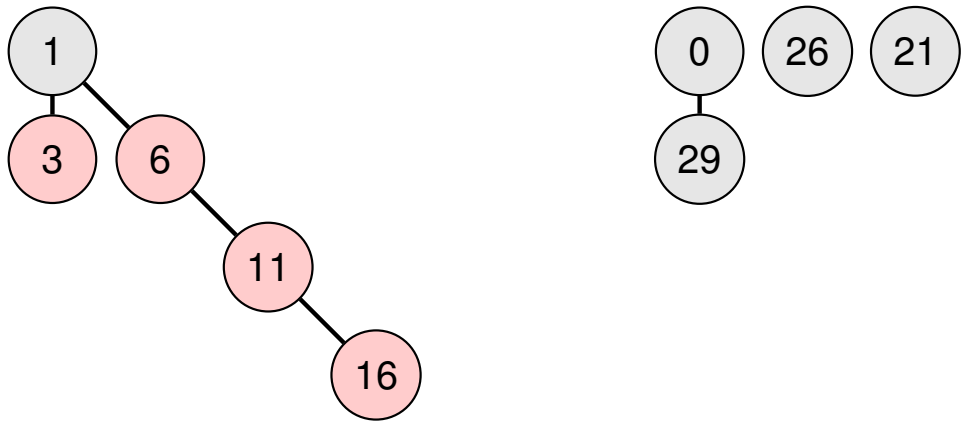
# Solution Microchallenge 5

Decrease-Key(28 $\rightarrow$ 0) Cascading Cut 1



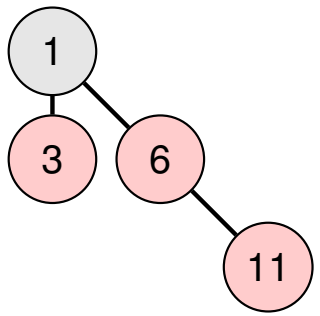
# Solution Microchallenge 5

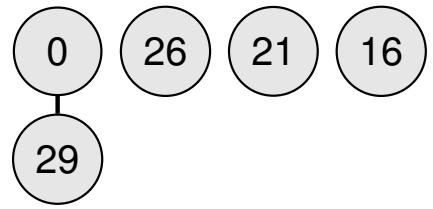
Decrease-Key(28 $\rightarrow$ 0) Cascading Cut 2



# Solution Microchallenge 5

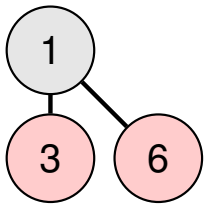
Decrease-Key(28 $\rightarrow$ 0) Cascading Cut 3

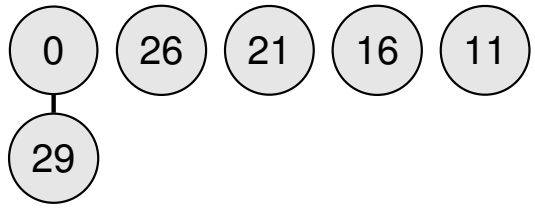


# Solution Microchallenge 5

Decrease-Key(28 $\rightarrow$ 0) Cascading Cut 4



# Solution Microchallenge 5

Decrease-Key(28 $\rightarrow$ 0) Cascading Cut 5