

## MST algorithms

~~OXO~~

### Reverse-delete

- sort edges by weight  $\rightarrow M \cdot \log M$
- in decreasing order  $\rightarrow m$  times
  - $\hookrightarrow$  remove edge if it doesn't disconnect

BFS  $M$

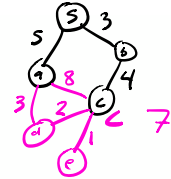
$$O(M \log M + M^2) = O(M^2)$$



```

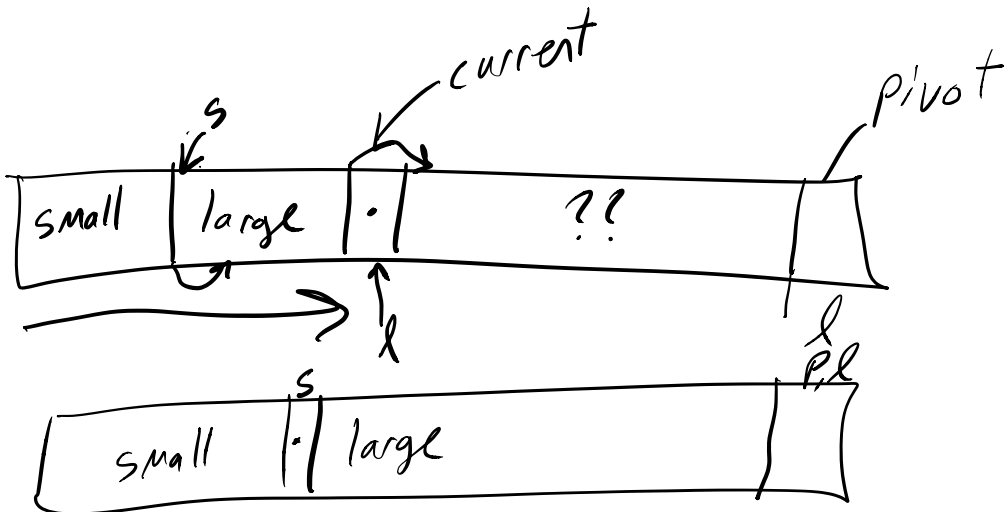
Function Dijkstra(src):
  visited_costs = new hash table
  frontier = new min priority queue
  add 0src to frontier
  While frontier not empty:
    cost_to_vertex = next priority in frontier
    vertex = remove from frontier
    If vertex not in visited_costs
      add vertex:cost_to_vertex to visited_costs
    For each neighbor of vertex:
      if neighbor not in visited_costs:
        total_cost = cost_to_vertex + edge(vertex,neighbor).cost
        add cost:neighbor to frontier
      End If
    End For
  End If
  End While
  return visited_costs
End Function

```



QuickSort (array, start, end)  
 if (start >= end)  
 return

*Pivot index* → P = partition (array, start, end)  
 QuickSort (array, start, P-1)  
 QuickSort (array, P+1, end)



Partition (array, start, end) {

  swap a random element w/end

  p = end

  s = l = 0

  for ( ; l < p ; l++ ) {

    if array[l] < array[p]

      swap(array, l, s)

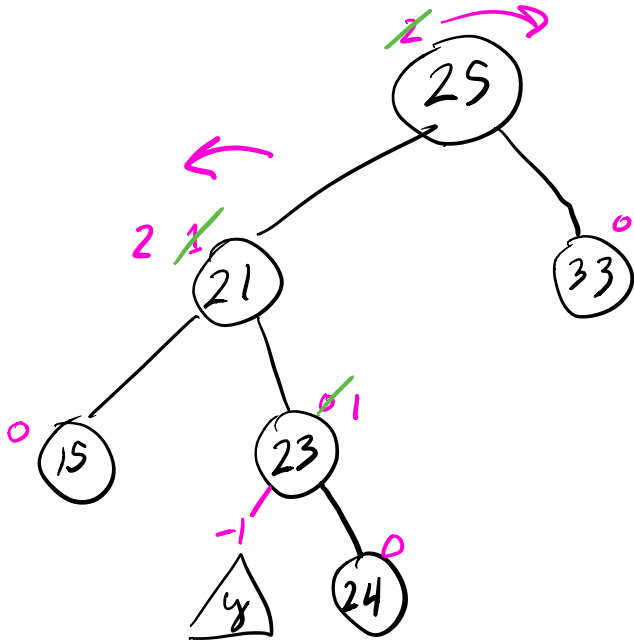
      s++

  }

  swap(array, p, s)

  return s

}



insert 24

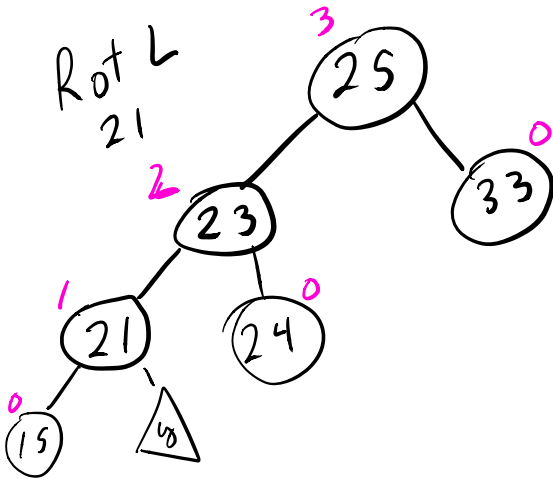
LR

Rot L on 21

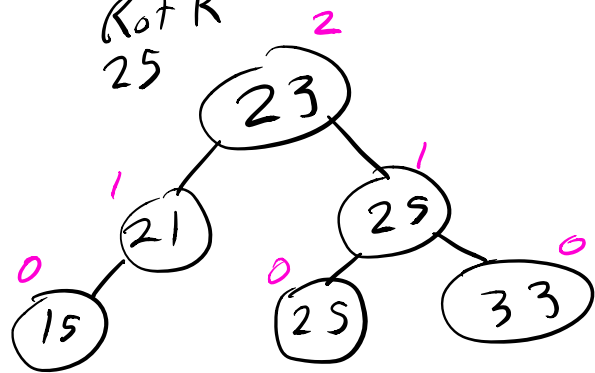
LL

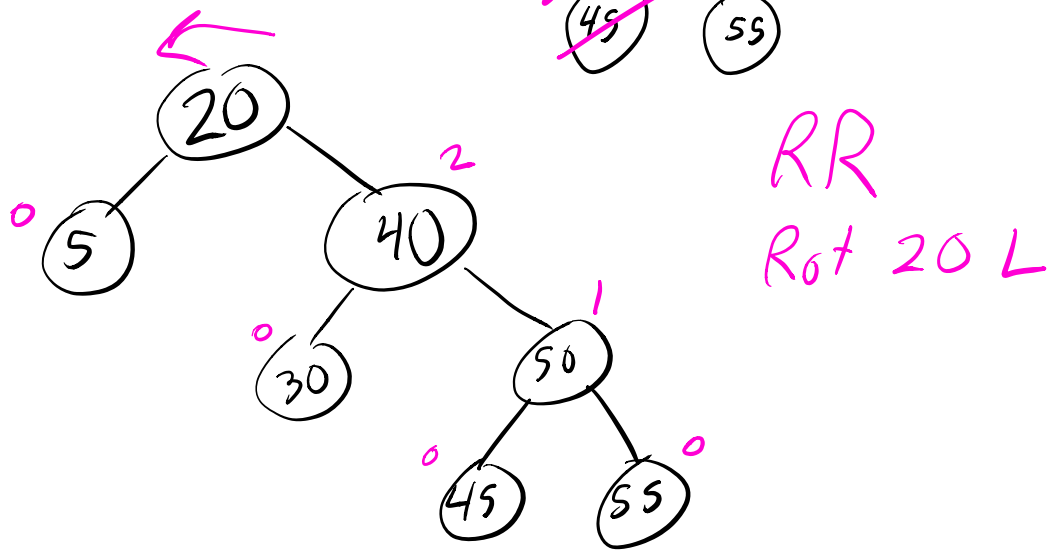
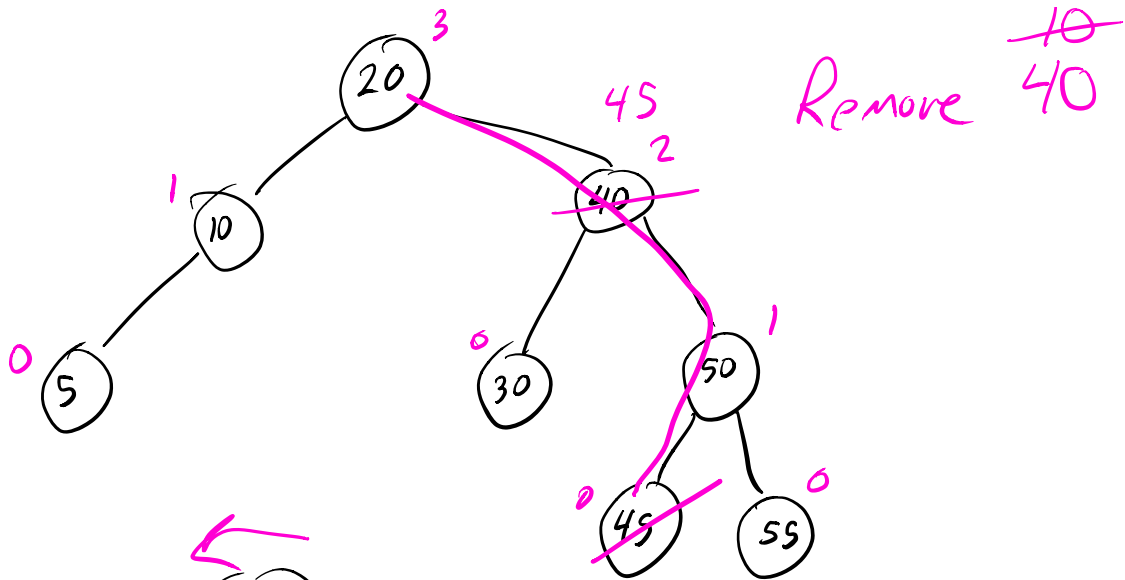
Rot R on 25

Rot L  
21

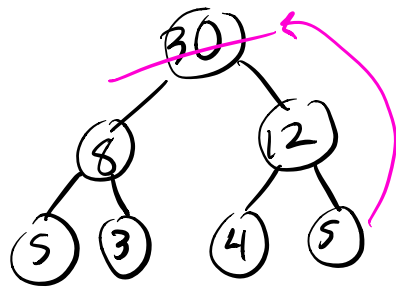
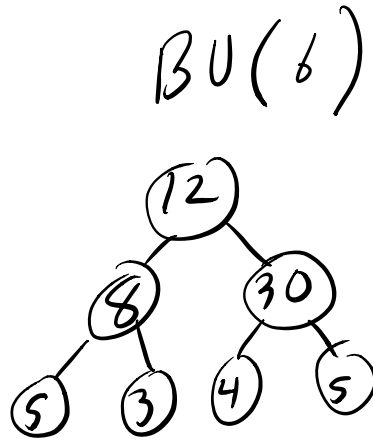
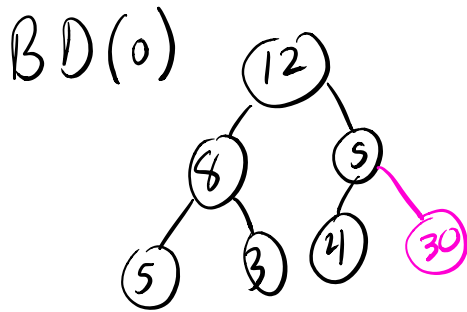
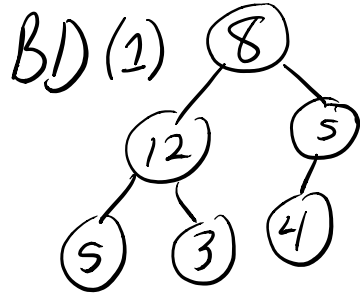
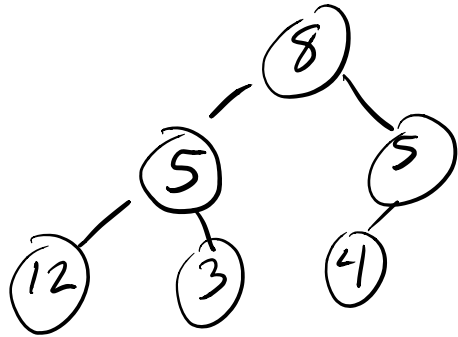


Rot R  
25

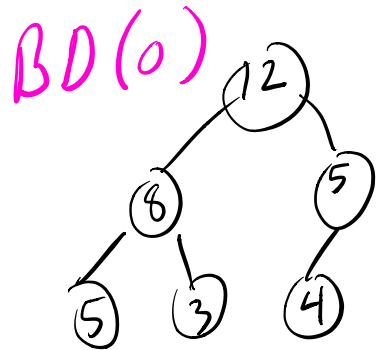




[ 8, 9, 5, 12, 3, 4, 30 ]



[ ~~30~~, 8, 12, 9, 3, 4, 5 ]



$$f(n) = \frac{n^2 + 1}{n} \text{ is } O(n)$$

$\exists c, k > 0$  st.

$$\forall n > k \quad f(n) \leq c \cdot g(n)$$

$$f(n) = \frac{n^2 + 1}{n} = \frac{n^2}{n} + \frac{1}{n} = n + \frac{1}{n}$$

w.t.s

$$c \cdot n \geq \underbrace{n + \frac{1}{n}}_{\forall n > k}$$

$$k=1$$

$$c=2$$

$$n \leq 1 \cdot n$$

$$\frac{1}{n} \leq 1 \cdot n$$

$$n + \frac{1}{n} \leq 2n \quad \forall n > 1$$