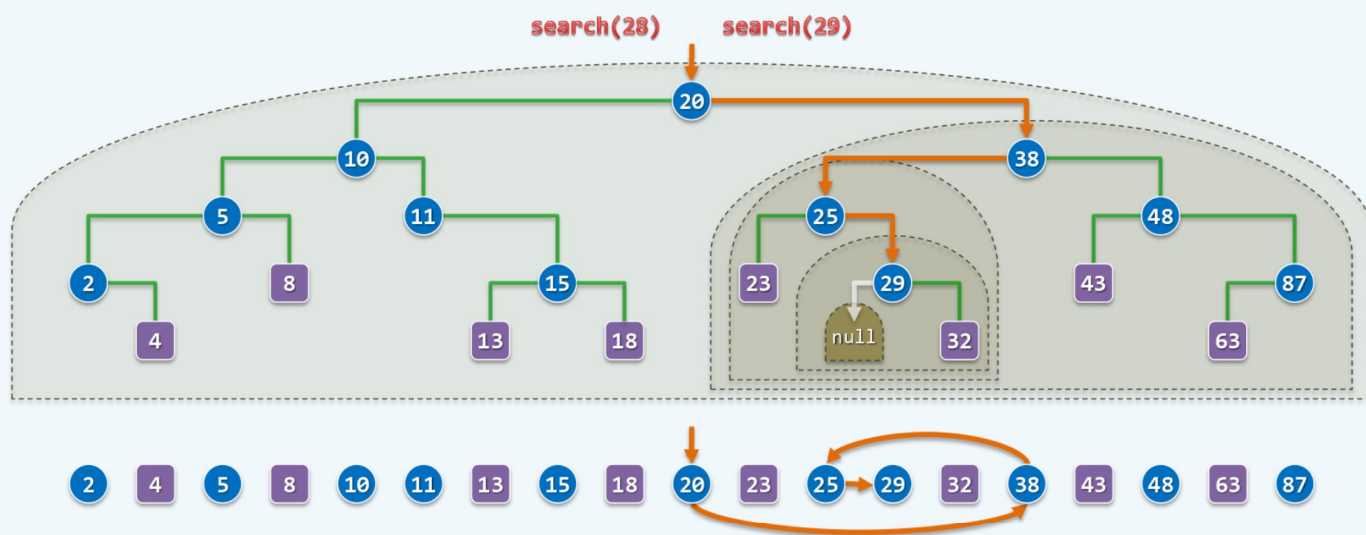


## BST & BBST



## Hashtable + Dictionary + Map

❖ `beauty = dict # Python dictionary (hashtable)`

( { "沉鱼":"西施", "落雁":"昭君", "闭月":"貂蝉", "羞花":"玉环" } )

`print beauty`

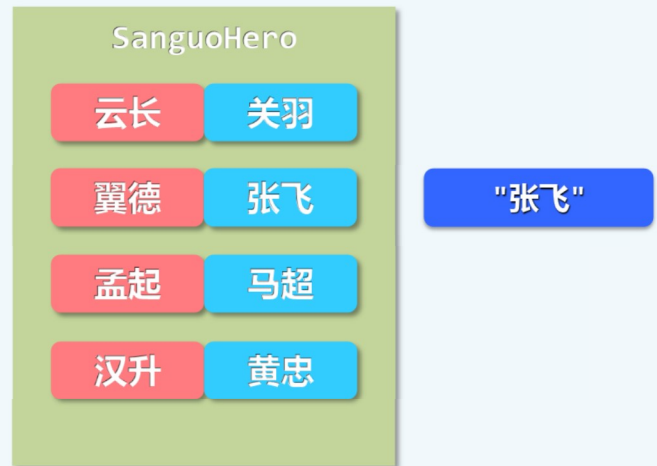
❖ `beauty["红颜"] = "圆圆"`

`print beauty`

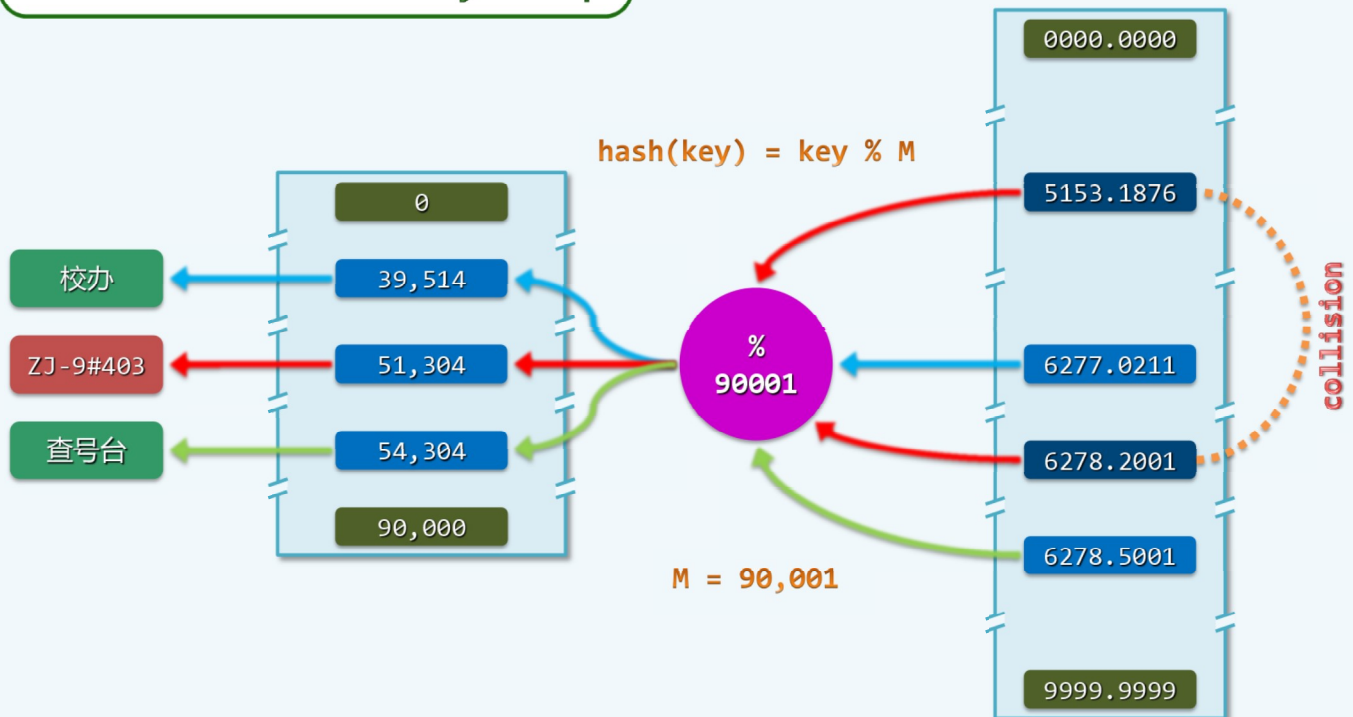
`get("翼德")`

❖ `for alias, name in beauty.items():`

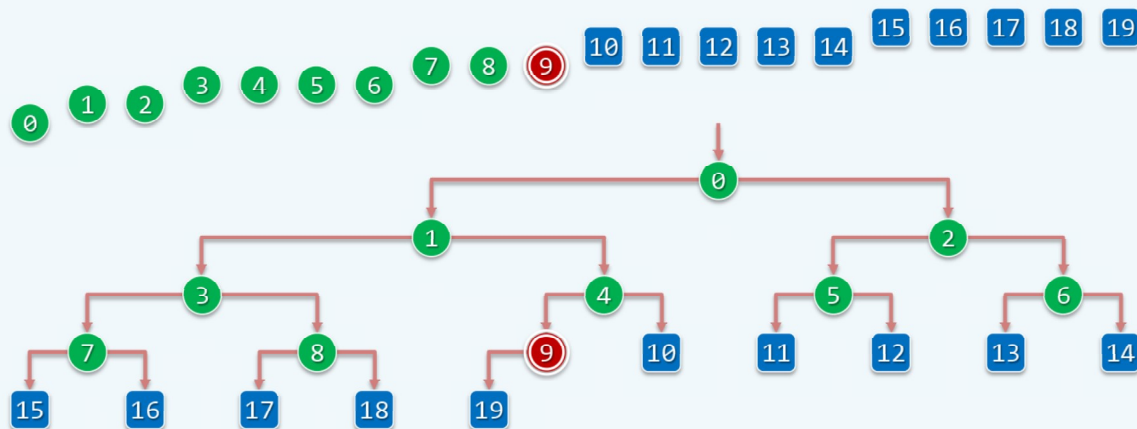
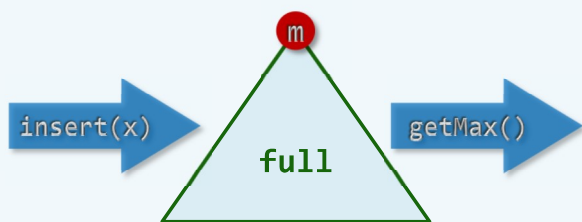
`print alias, ":", name`



## Hashtable + Dictionary + Map



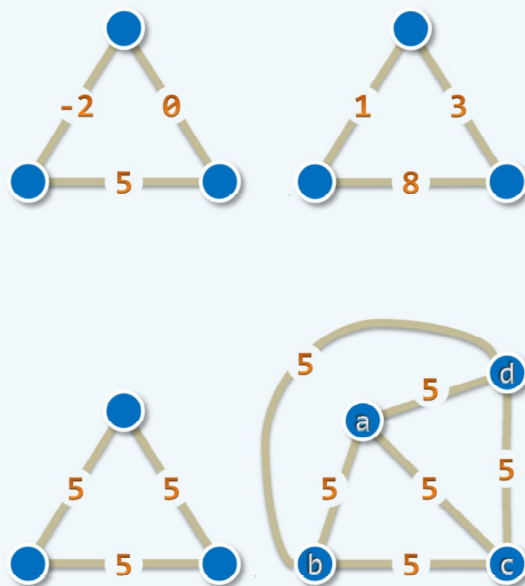
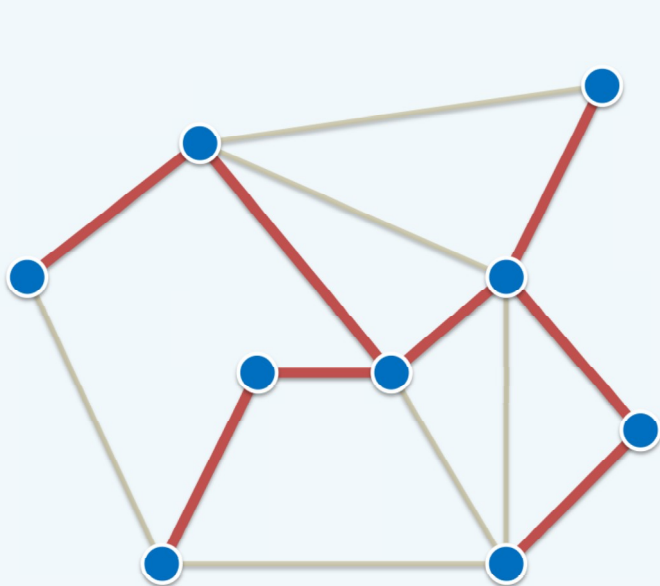
## Priority Queue ~ Heap



Tsinghua Algorithm Camp (Spring 2018), Junhui DENG

4

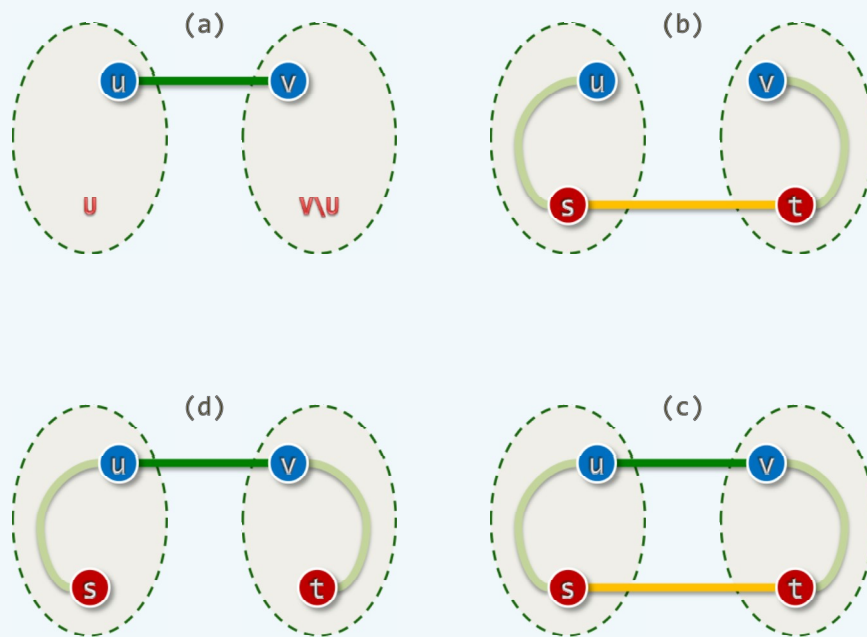
## Minimum Spanning Tree



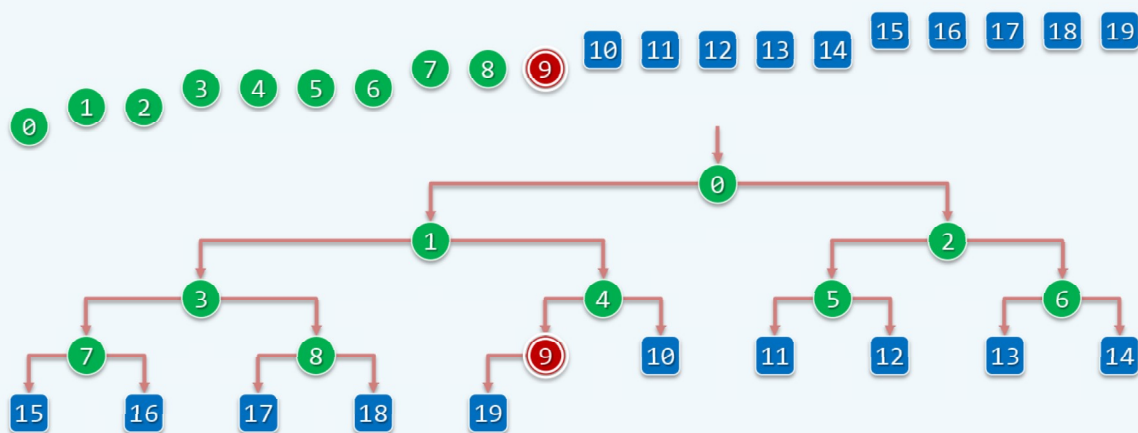
Tsinghua Algorithm Camp (Spring 2018), Junhui DENG

5

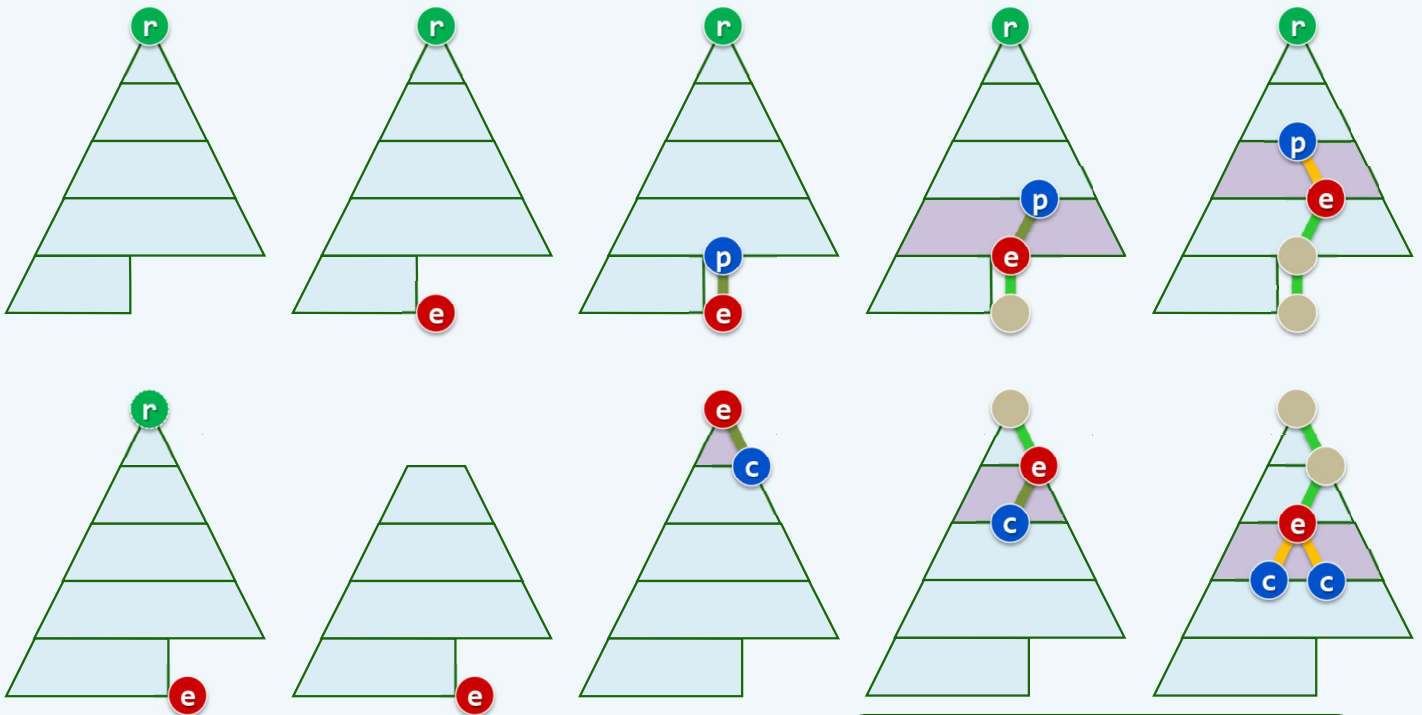
## Prim: Cut + Cross



## Prim: Heap



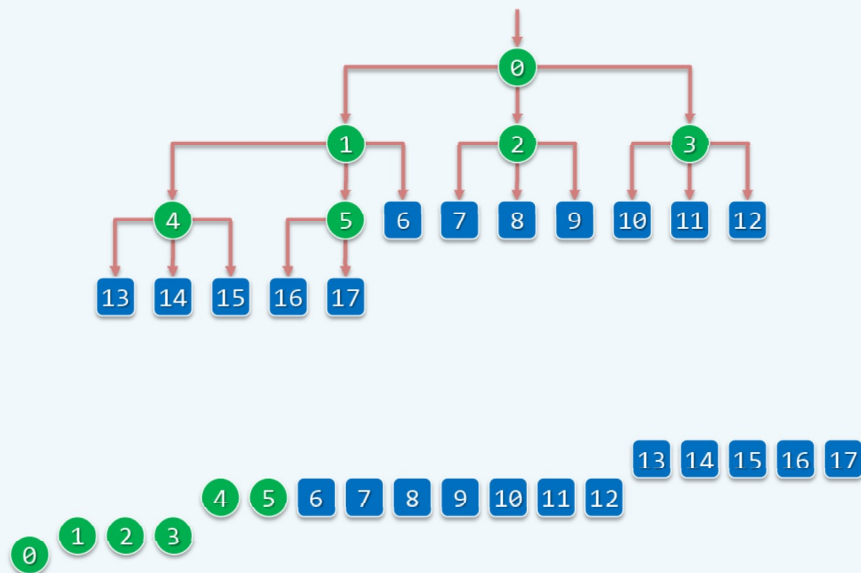
## Prim: Heap



Tsinghua Algorithm Camp (Spring 2018), Junhui DENG

8

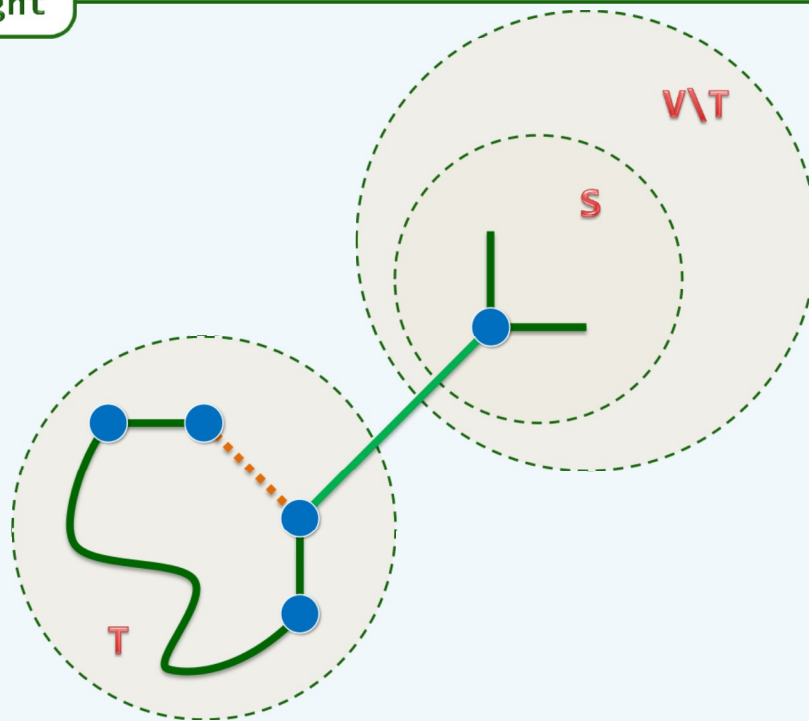
## Prim: d-Heap



Tsinghua Algorithm Camp (Spring 2018), Junhui DENG

9

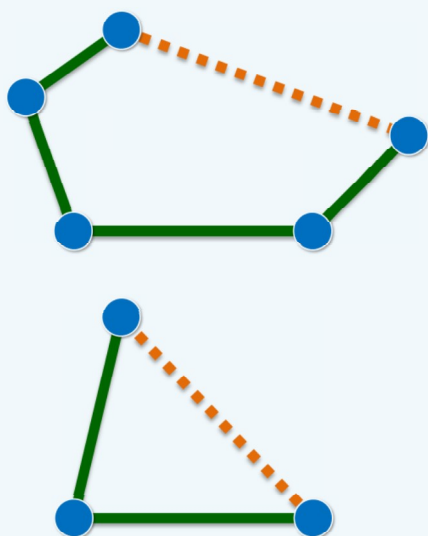
## Kruskal: Sorting by Weight



Tsinghua Algorithm Camp (Spring 2018), Junhui DENG

10

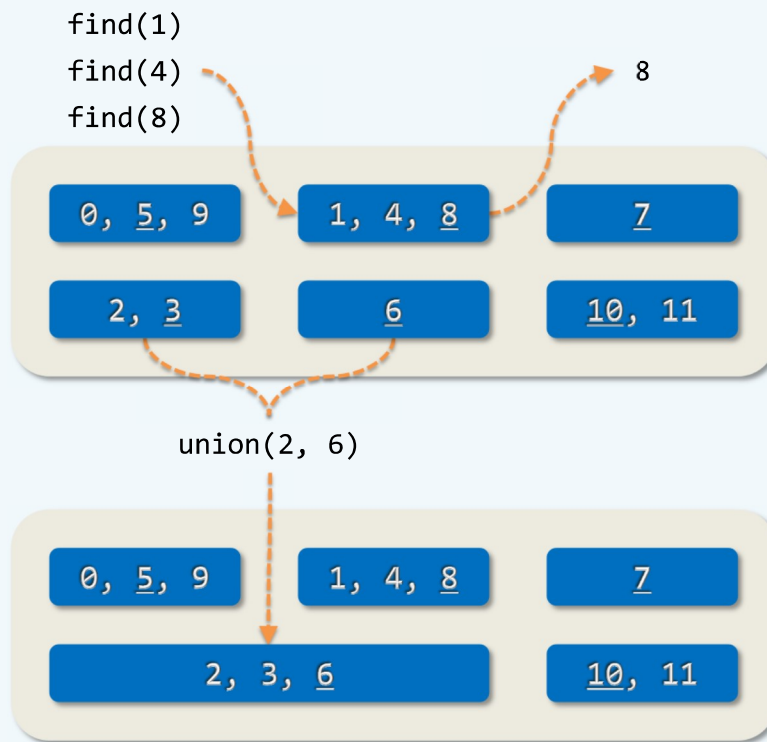
## Kruskal: Loop Detection



Tsinghua Algorithm Camp (Spring 2018), Junhui DENG

11

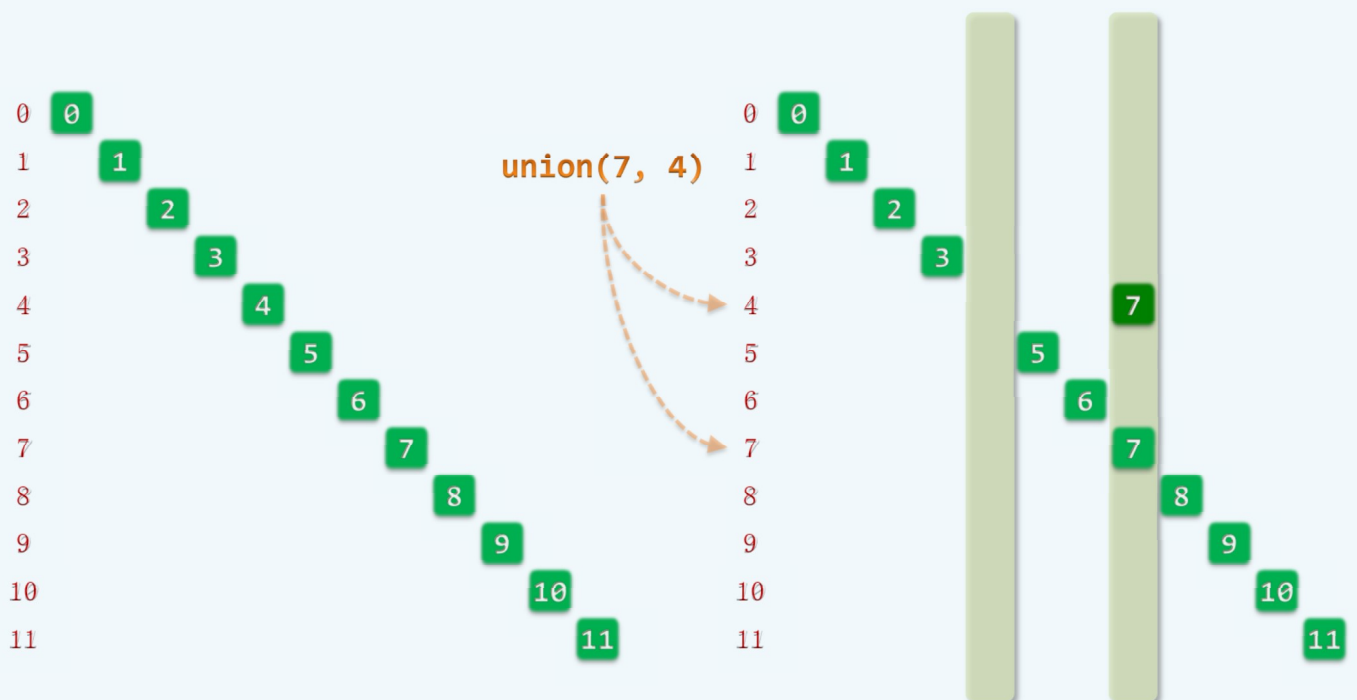
## Union-Find



Tsinghua Algorithm Camp (Spring 2018), Junhui DENG

12

## Quick-Find

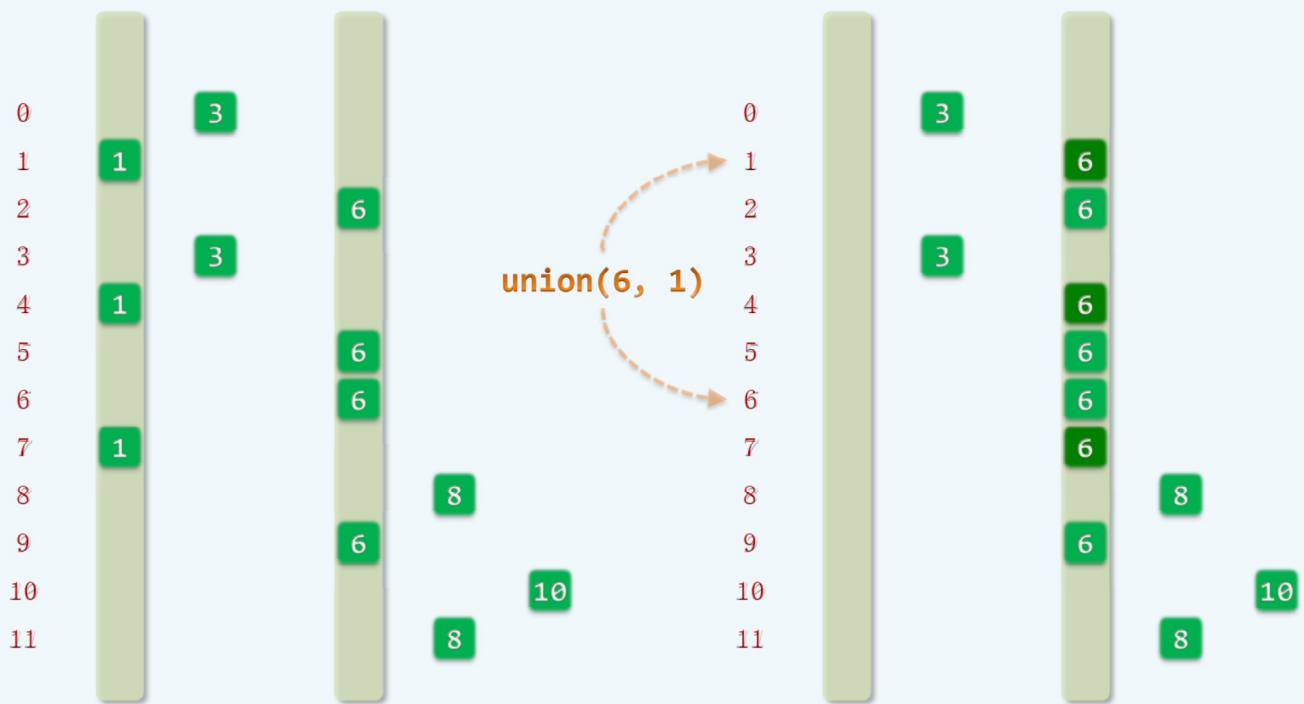


Tsinghua Algorithm Camp (Spring 2018), Junhui DENG

13



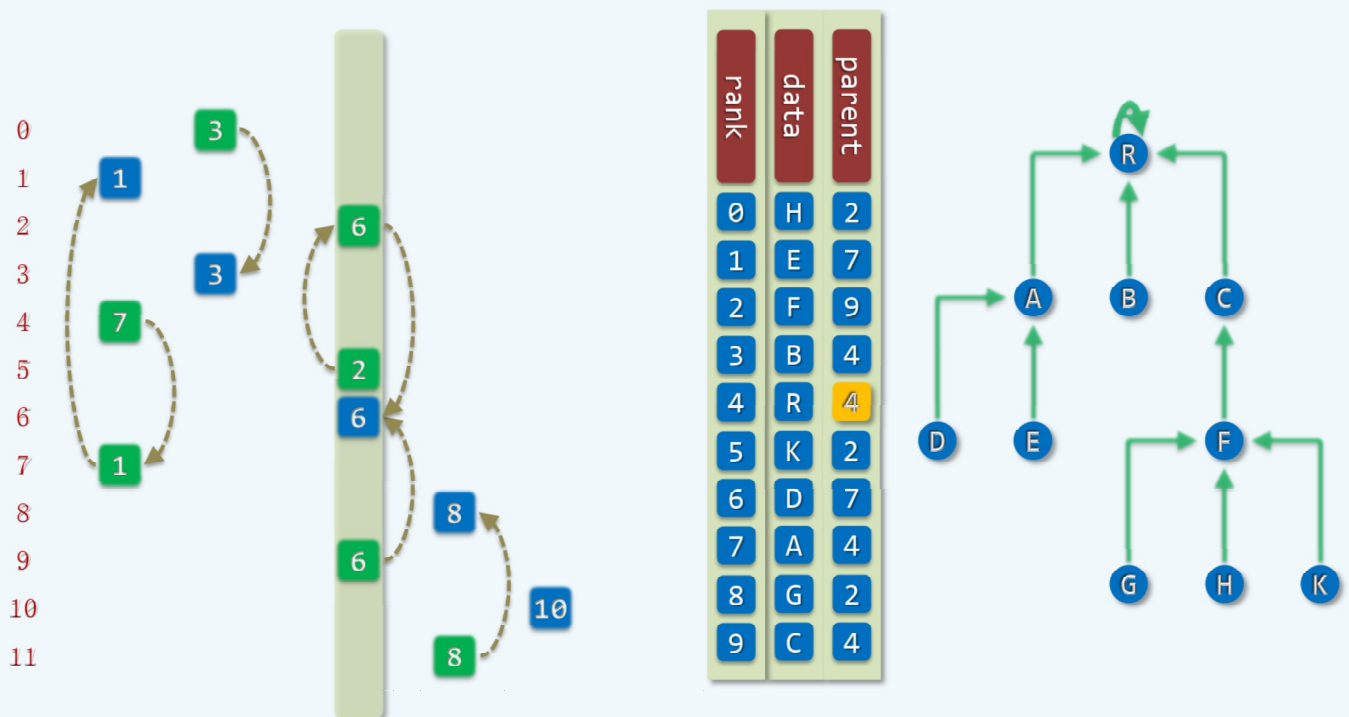
## Slow-Union



Tsinghua Algorithm Camp (Spring 2018), Junhui DENG

14

## Quick-Union

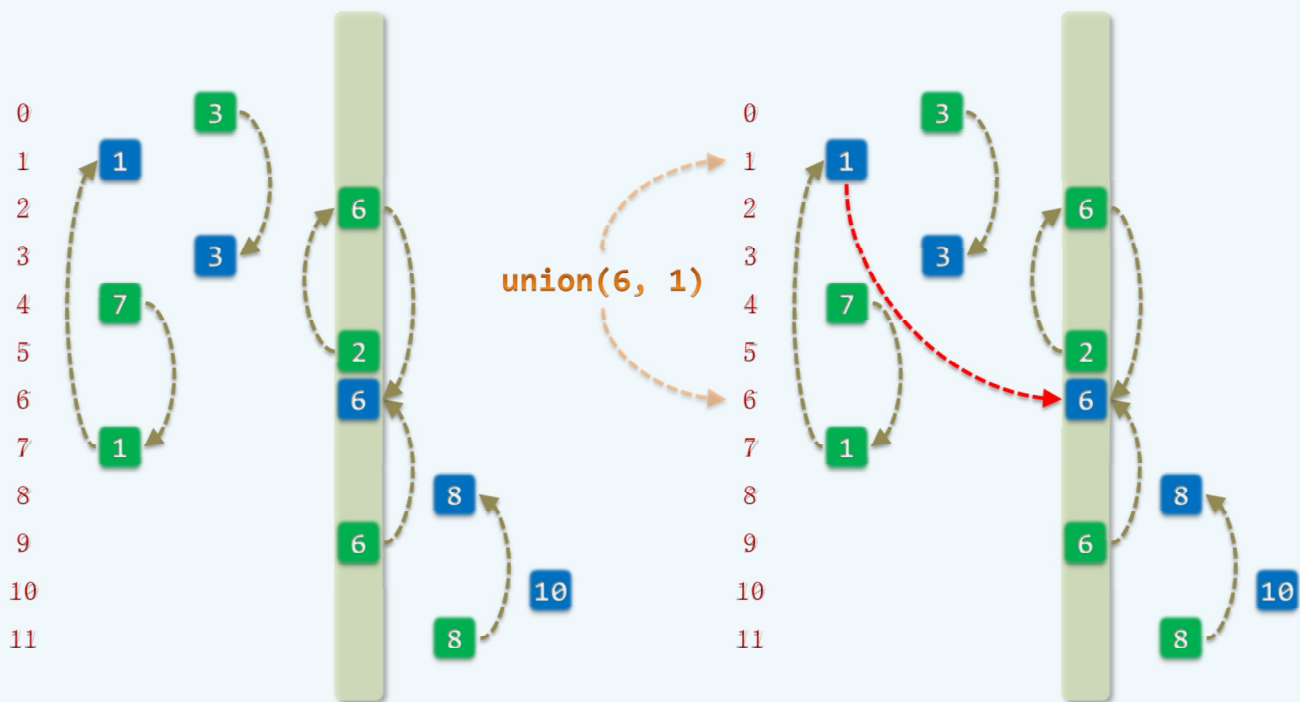


Tsinghua Algorithm Camp (Spring 2018), Junhui DENG

15



## Quick-Union



## Path Compression

