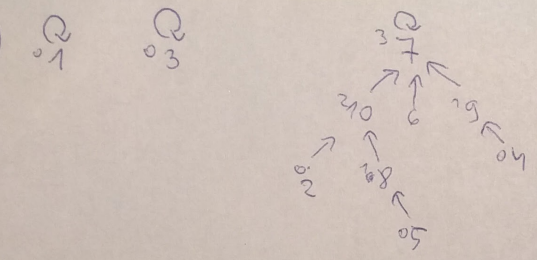
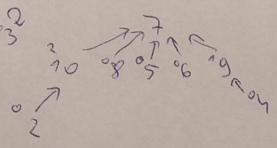
3) a)



FIND 3 : ISTO

FIND 5 :



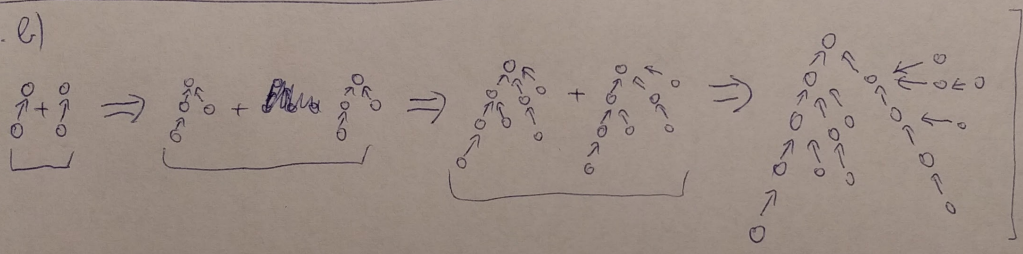
FIND 6 ISTO

FIND 9 ISTO

FIND 10 ISTO

---

3. b)



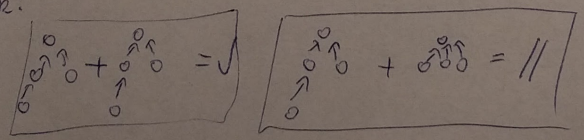
$h = 4$, ELE = 16

$\lg(16) = 4$

NAJSLABŠI ČAS = LOG

ČE ZDRUŽUJEMO ENAKA DREVESA TAKO DA KOREN NEKEGA DREVESA PRIDRUŽIMO DRUGEGA, NE BO PRIŠLO DO NOBENE KOMPRESIJE POTI MED GRADNJO IN S TEM BOMO DOSEGLI NAJSLABŠI ČAS.

ZDRUŽUJEMO PA DVA ENAKA, W NE ENEGA IZRODNEGA, DRUGEGA PA POPOLNEGA ZARADI SLABŠIH PERFORMANC. PRIMER:



```
class DisjunctiveSet:
    parent = dict()                                          #stores parent for every node

class Node:
    children_heights = []
    parent
    key

def find(x, height):

    max_height = 0
    if len(children_heights):
        max_height = max(children_heights) + 1               #plus self


    if (parent[x] != x): #while not root

        parent[x].children_heights.append(max_height)        #apends height of this node
        parent[x], max_height = find(parent[x], height);     #recursevly iterates till root

    return parent[x], max_height;
```