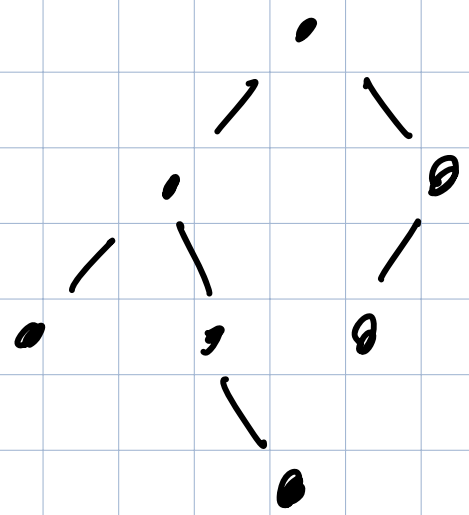
Velkommen til IN2010 uke 2

Balanserte søketrær



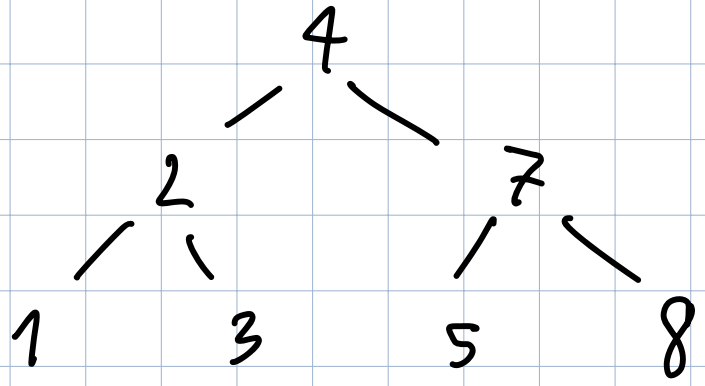
- Luna (arthuk@uio.no)

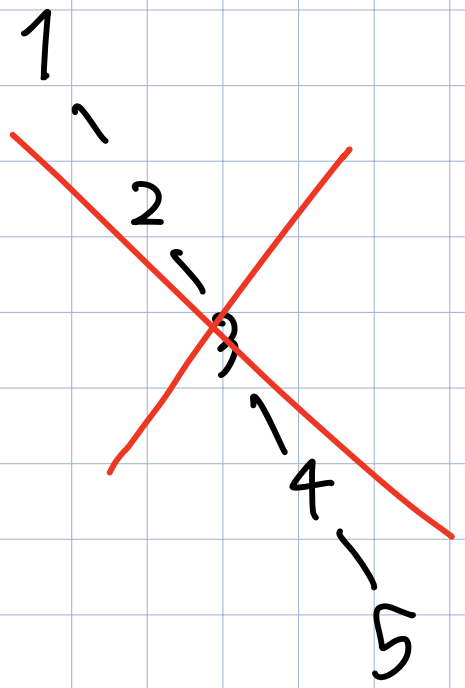
Plan for timen

- AVL-trær
- stdin/stdout
- Rød-svarte trær?

AVL-trær

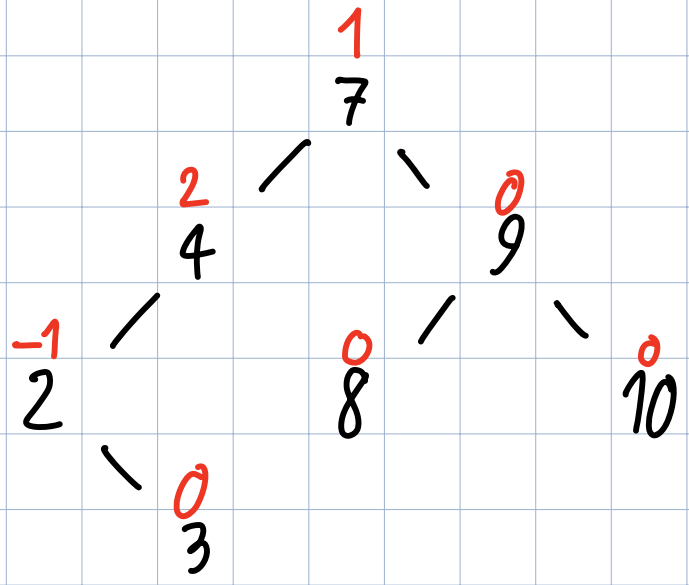
- Selvbalanserende binært søketre
- $O(\log n)$ på oppslag, innsetting og sletting

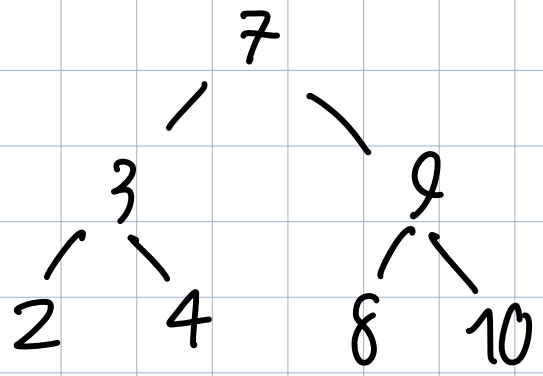




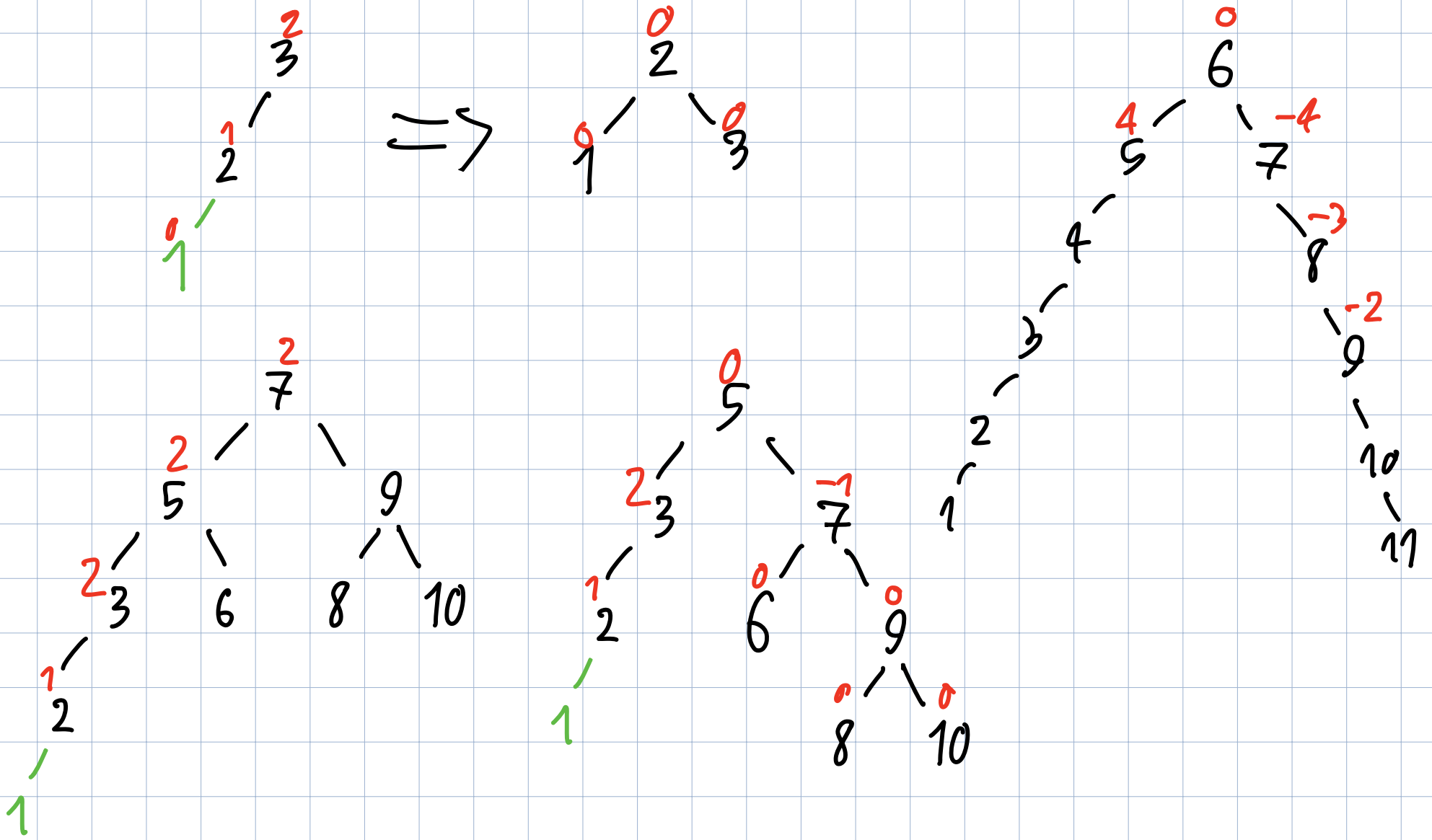
Balansefaktor

- Høyden av venstre subtre minus høyden av høyre subtre
- Egenskap ved et tre
- Hver node er roten av sitt eget tre, så hver node har en BF

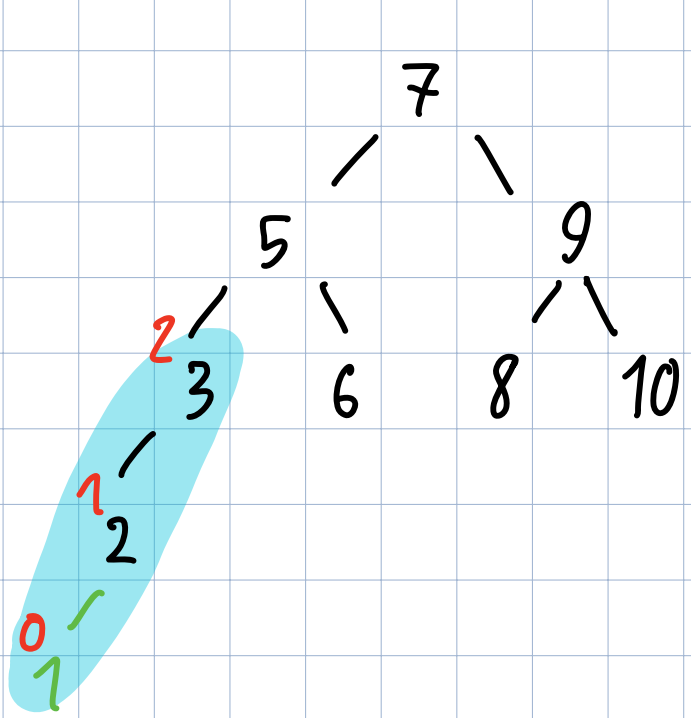


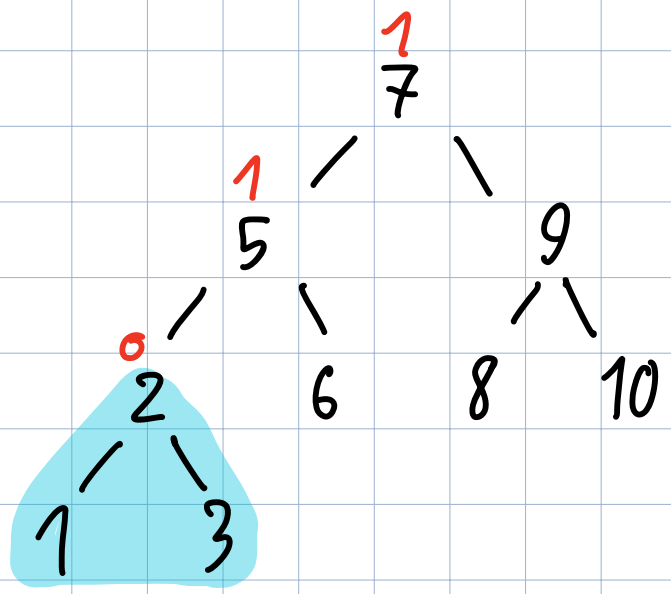


- Hvis et tre har blitt for venstretungt (f.eks. etter en innsetting), kan vi gjøre en høyrerotasjon

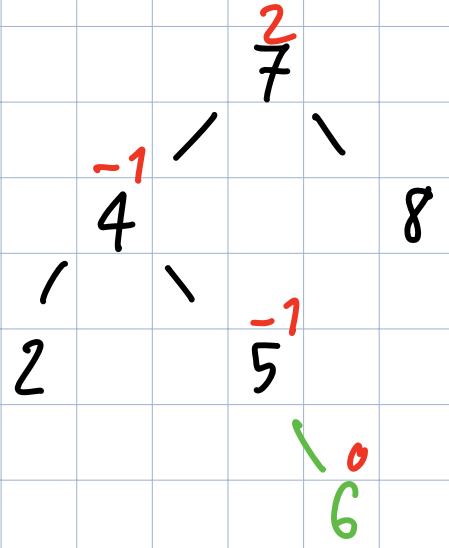


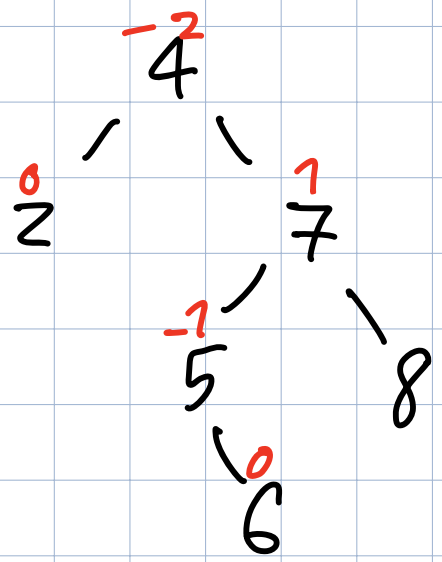
- Vi må gjøre rotasjonene på vei opp

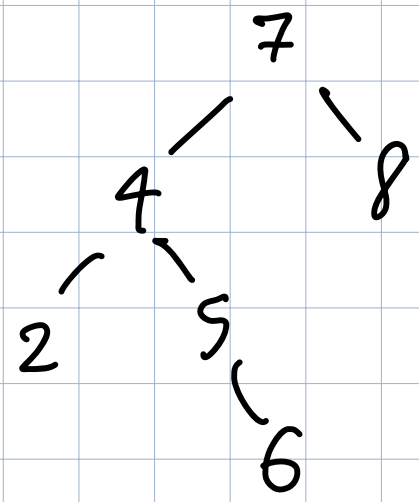


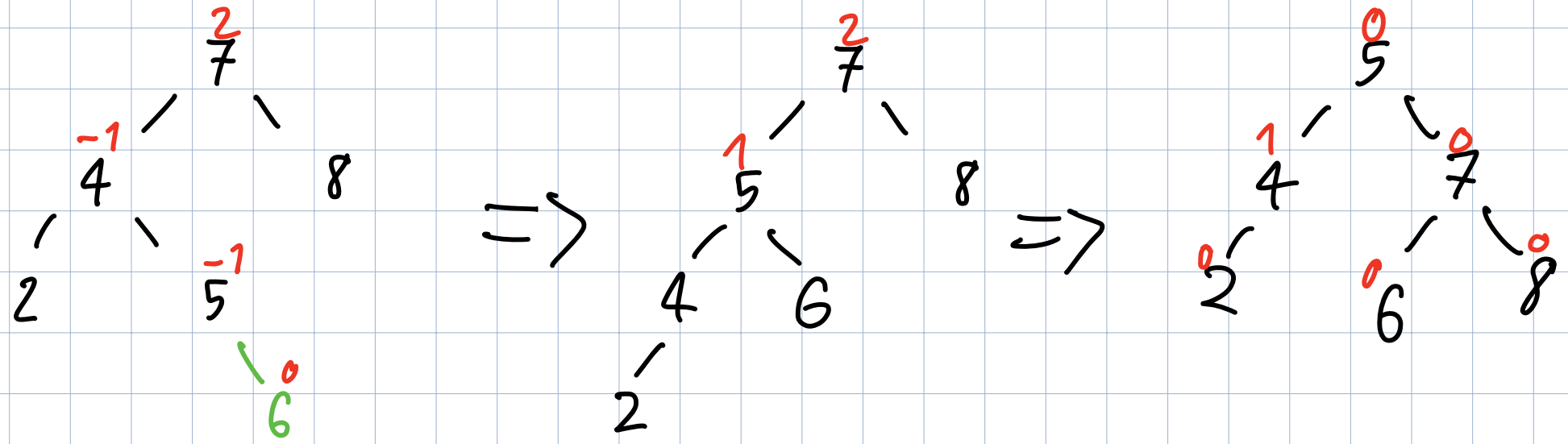


– Men vi trenger et siste triks i tillegg

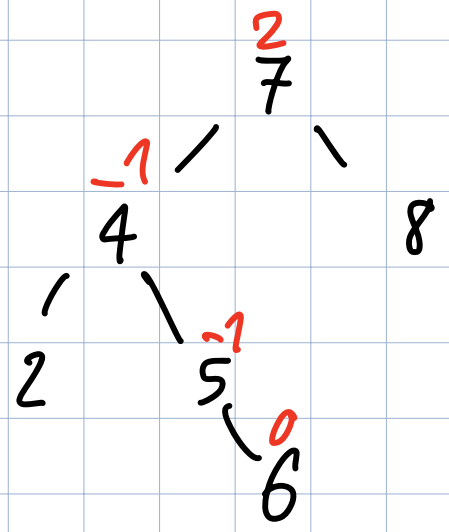







2
7
-1
4
8
2
-1
5
0
6
=>
2
7
1
5
8
4
6
2
=>
0
5
1
4
0
7
0
2
0
6
0
8

```
def insert(tree, n):
    if tree = null:
        return nyt + tre med n som rot
    if n < tree.elem:
        tree.left ← insert(tree.left, n)
    if n > tree.elem:
        tree.right ← insert(tree.right, n)
    return Balance(tree)
```




```
def Balance(tree)
    if BF(tree) > 1:
        if BF(tree.left) < 0:
            tree.left ← LeftRotate(tree.left)
        return RightRotate(tree)
```

Rød-svarte trær

- Roten er alltid svart
- En rød node har aldri et rødt barn
- Enhver sti fra roten til et tomt tre inneholder like mange svarte noder

