IN2010 uke 3-4

Jakob Hansen

15. september 2020

Ting vi kan snakke om i dag

- Repetisjon
- Balanserte trær
  - AVL trær
  - Rødsvarte trær
- Køer, heaps, huffman coding
- Obligen

Balanserte trær

- Hva er et balansert tre?
- Et tre der høyden er relativt lav i forhold til antall noder.
- Hvorfor ønsker vi balanserte trær?
- For eksempel for raskere søk og innsetting.

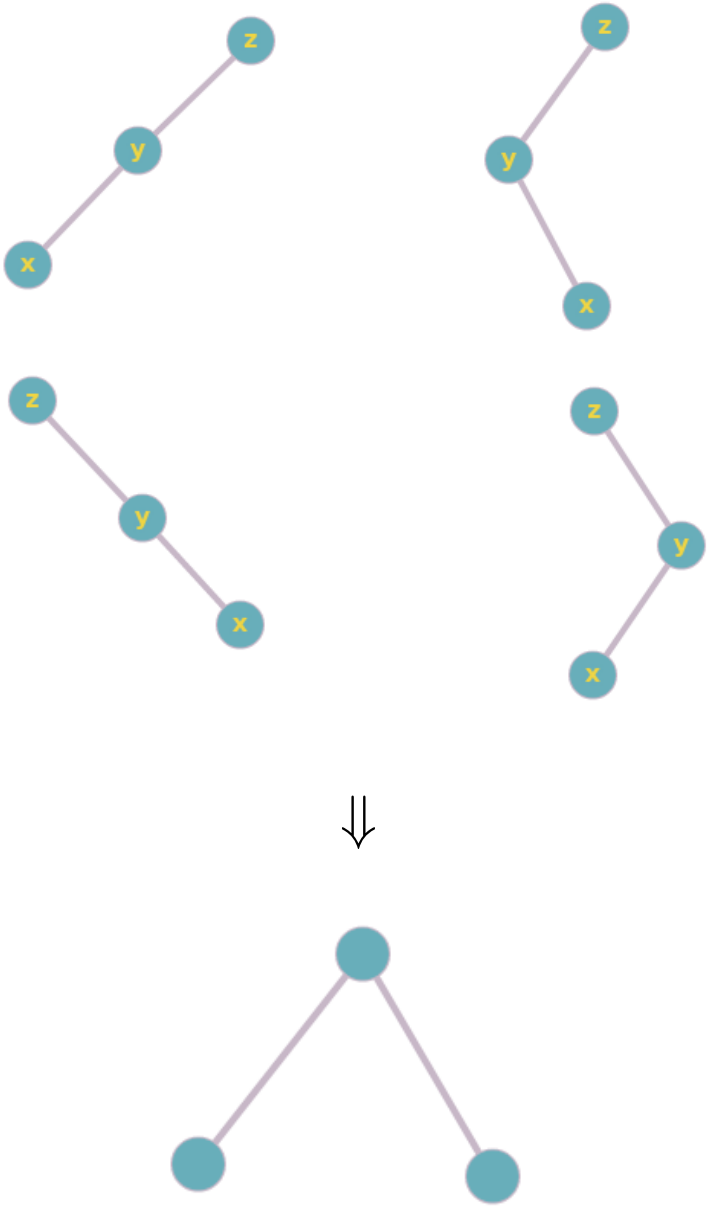
AVL trær

- Selvbalanserende binært søketre
- Hver node har en høydeverdi og en balanseverdi
- hoyde = max(venstrenode.hoyde, hoyrenode.hoyde) + 1
- hoyde(null) = -1
- balanse = venstrenode.hoyde - hoyrenode.hoyde
- balanse b til enhver node må være -1 $\leq$ b $\leq$ 1

Innsetting i AVL trær

- Sett inn i et AVL tre slik som et vanlig binært søketre.
- Oppdater høyde på vei oppover i rekursjonsstacken, og sjekk balanse.
- Hvis balansen er større enn 1 eller mindre enn -1, må vi fikse treet.
- Fiks treet ved å gjøre rotasjoner i forhold til der treet er "tyngre"

Rotasjoner



Sletting i AVL trær

- Nesten helt likt
- Slett som i et vanlig binært søketre
- Oppdater høyde på vei oppover i rekursjonsstacken og sjekk balanse
- Hvis balansen er større enn 1 eller mindre enn -1, må vi fikse treet.
- Gjør rotasjoner i forhold til der treet er “tyngre“

Rødsvarte trær

- Ikke grundig gjennomgang!
- Regler:
  - Alle noder er enten svarte eller røde
  - Hvis en node er rød, så er barna svarte
  - Alle stier fra roten til en nullpeker må inneholde like mange svarte noder
  - Ekstra: Roten er svart, nullpekere teller som svarte

Abstrakte datatyper

- Ofte definert som interfaces, med flere mulige implementasjoner
- Prioritetskø kan implementeres med array, heap, osv...
- Det som skiller implementasjonene er kompleksiteten til operasjonene.

Heap

- Tre-basert datastruktur som skal minne om en haug.
- MinHeap: Minste element øverst, bare større elem under.
- MaxHeap: Største element øverst, bare mindre elem under.
- Kan implementeres som et tre (noder, barn) eller et array?? :O
- Kompleksitet?
- insert? -> O(log(n))
- removeMin? -> O(log(n))

Huffman coding

- “Encode“ strenger så effektivt som mulig (få bits)
- Mål: Bokstavene som dukker opp oftest, skal bruke så få bits som mulig.
- Ekstra: Uniquely decodable.