

Trær:

Terminologi:

- Node (node / vertex (flertall: vertices))
- Kant (edge)
- Sti (path)
- Rot (root)
- Kun *en* sti mellom alle noder (også via rota), ellers er det en graf
- Over - under
- Mor/forelder (parent) - barn (child) - barnebarn - søsken - besteforeldre
- Intern node (ikke-terminal/-bladnode) - har *minst ett* barn
- Blad-/terminalnode (ekstern node) - har ingen barn, nederst på en sti
- Grad - antall barn hver node har (*treets* grad – noden med høyeste grad)
- Nivå - rot er på nivå nr.0 (eller 1)
- Dybde - antall kanter fra *rota* og til *en gitt* node (jfr. nivå)
- Høyde - antall kanter fra *en* node og ned til dens fjerneste bladnode
- Sti-lengde - summen av *alle* stiene fra rota og til *enhver* node
- Ordnet/sortert vs. ikke-sortert
- Subtre
- Skog
- Multiveis tre - hver har opptil N barn – evt. fyller ut med bladnode(r)
- Binært tre - to barn (venstre/left og høyre/right) - enkleste form for multiveis-tre
- Fullt binært tre - *alle interne* noder har *eksakt* to barn (men treet trenger ikke å være balansert av den grunn)
- Komplette binært tre - alle bladnodene slutter på samme nivå, unntatt evt. det aller nederste, men det er fylt opp med noder fra venstre mot høyre

Senere: Binært søketre, B-tre, Balansert tre, 2-3-4 tre, Red-Black tre

Noen regler/egenskaper:

1. *Eksakt en* sti (via rota?) mellom to noder i et tre
2. N noder har N-1 kanter (fordi alle med *en* link opp til mor, unntatt rota)
3. Binært tre: N noder har N+1 tomme «barn»/null(ptr)
4. *Fullt balansert* binært tre: høyde er ca. $\log_2 N$ (avrundet opp til nærmeste heltall)

Traversering av trær – primært 4x måter:

1. Preorder:

- **visit (seg selv)**
- Traverser venstre
- Traverser høyre

2. Inorder:

- Traverser venstre
- **visit (seg selv)**
- Traverser høyre

3. Postorder:

- Traverser venstre
- Traverser høyre
- **visit (seg selv)**

4. Levelorder:

- **visit alle** på ett og ett nivå, fra venstre mot høyre, ovenfra og nedover (dvs. «les treet linjevis»)