

AlgDat

Overblikk

Pensum

- <http://www.ia.hiof.no/algdat/pages/pensum/index.html>
- <http://www.ia.hiof.no/algdat/pages/forelesningsplan/index.html>
- <http://www.ia.hiof.no/algdat/pages/oppgaver/index.html>

Datastrukturer

- Ustruktert samling: Ingen "ordning"
- Lineære strukturer (Lister)
 - Neste / forrige
- Hierarkiske (trær)
 - Foreldre/barn
 - Søk/traversering:
 - Dybde (pre, in, postorder: rekursivt),
 - Bredde (levelorder: kø)
 - Implisitte trær (tabellform)
- Nettverk:
 - Naboer
 - Søk/traversering:
 - Dybde (rekursivt)
 - Bredde (kø)

Tabeller (arrays)

- Endimensjonale
- Matriser
- Iterasjon
- Finne, sette inn, slette
- Best til statiske anvendelser

Lenkede strukturer

- Lenkede lister
- Trær
- Nettverk
- Iterasjon, traversering
- Finne, sette inn, slette
- Best til *dynamiske* anvendelser

Stack

- Kun interesserte i "toppen"
- Peek, push, pop
- Bevarer "historikk"
- Syntakssjekking
- Håndtering av f.eks. sammensatte binæruttrykk
- "Ekvivalent" med rekursjon

Kø

- Kun interesserte i "først" og "sist"
- Peek, enqueue, dequeue
- Mange "naturlige" anvendelser

Prioritetskø

- Kun interesserte i "største"/"minste"
- Peek, enqueue, dequeue
- Mange "naturlige" anvendelser

Trær

- Binære trær (0, 1 eller to barn)
- Binære søketrær
- Balanserte trær (AVL)
- Binærheap

Hashtabell

- Key/value
- Hashfunksjon som transformerer Key til en indeks i en tabell
- Hvis vi sørger for at tabellen ikke er mer en halvfull, vil vi i gjennomsnitt bruke ca 1.6 oppslag

Algoritmer

- Iterasjon
 - For, while, do while
 - Tabeller: Indekser
 - Lenkede strukturer: referanser
- Traversering: Lister, Trær, Nettverk
 - Dybde
 - Bredde
- Rekursjon
- Binærsøk
- Sortering
- AVL balansering
- Permutasjoner
- Splitt og hersk
- Problemtransformasjoner

Algoritmeanalyse

- Big-Oh, asymptotisk analyse, worst case.
- Klasser:
 - Konstant
 - Logaritmisk
 - Lineær
 - Loglineær
 - Kvadratisk, kubisk
 - Eksponentiell
- Matematisk
- Kode
- Empiri