

# Notater Algdat 2018

Sebastian Ellefsen

August 27, 2018

## 1 Forelesing 2: Datastrukturer

- $\sum_{i=0}^{n-1} i = \frac{n \cdot (n-1)}{2}$
- $\sum_{i=0}^{h-1} 2^i = 2^h - 1$

## 2 Stack & Queue

### 2.1 Stack

Kun adgang øverst (legge til, trekke fra). Først inn, sist ut (LIFO)

Metoder:

- STACK-EMPTY(S) // Sjekk om stakken er tom
- PUSH(S,  $x$ ) // Legg til i stakken
- POP(S) // Fjern fra toppen av stakken

### 2.2 Queue

Først inn, først ut (FIFO). Implementeres som ringbuffer (modulært)

Metoder:

- ENQUEUE(Q,  $x$ ) // Legge til i køen
- DEQUEUE(Q) // Hente ut første element i køen

## 3 Linked List

- Noder som peker til neste (noen ganger forrige, dobbeltlenket - "doubly linked list").
- Tar lineær tid å slå opp på en gitt posisjon.

- Tar konstant tid å sette inn/fjerne elementer.

Metoder:

- LIST-SEARCH( $L, k$ ) // Søk etter ett element/nøkkel  $k$  i listen  $L$
- LIST-INSERT( $L, x$ ) // Sette inn en node  $x$  i listen  $L$
- LIST-DELETE( $L, x$ ) // Slette en node  $x$  fra listen  $L$

## 4 Hashtables

- Både lesing og skriving (slå opp og sette inn / fjerne) tar konstant tid
- Direkte adressering
- Modifisert nøkkel er indeks

### 4.1 Chaining

Hashtabell sammen med linket liste. Nøkkelen settes inn som en node som linker videre til andre nøkler med samme hash.

Forventet listestørrelse er en konstant (summen av mindre og mindre brøker).

## 5 Dynamic tables

Når en liste begynner å bli full re-allokerer man til en større liste.

Amortisert arbeid: Gjennomsnittlig kjøretid per operasjon etter mange operasjoner.

Avg. case : Snitt over instanser

Amotisering : Snitt over operasjoner

## 6 Misc.

Kapittel 10..3 er selvstudium