



# Perspektiverende Datalogikursus

## Algoritmer og kompleksitet

Gerth Stølting Brodal

# Perspektiverende kursus

Formål:

- Vise bredden af Datalogi.
- Vise fagets anvendelighed.
- Vise konkrete eksempler på hvad datalogi er.
- Variation af undervisningsformen.

# Perspektiverende kursus

## Formål:

- vise bredden af Datalogi.
- vise fagets anvendelighed.
- vise konkrete eksempler på hvad datalogi er.
- variation af undervisningsformen.

## Form:

Onsdag: 1-2 timers forelæsning (denne uge: 1 time).

Torsdag: 7 timers praktiske øvelser, "Open Learning Center".

Fredag: 1-2 timers forelæsning (denne uge: 2 timer).

# Perspektiverende kursus

- Uge 1: Algoritmer og kompleksitet
- Uge 2: Formelle sprog og web-teknologi
- Uge 3: Modellering og simulation af distribuerede systemer
- Uge 4: Eksperimentel systemudvikling
- Uge 5: Algoritmisk spil-teori
- Uge 6: Kryptologi og datasikkerhed
- Uge 7: Dynamisk OO-programmering



# Algoritmer og kompleksitet

Algoritme: Klart beskrevet metode til løsning af en opgave.

# Algoritmer og kompleksitet

Algoritme: Klart beskrevet metode til løsning af en opgave.

Eksempler:

2 dl havregryn  
4 dl vand  
Hæld alt i gryde.  
Kog 3 min.  
Smag til med salt.

**Madopskrift**

50-35-30 g Tvinni  
to-trådet grøn  
Pinde nr. 3

Slå 38-28-20 m op,  
strik 4-3-3 p glatstr,  
start med r p. Lav  
raglan-indtag 2 r 2  
dr r sm.

**Strikkeopskrift**

```
int i,k;  
for (i=0;i<N;i++){  
    A[i] = B[i++];  
    k = k+i;  
}
```

**Computerprogram**



# Algoritmik

**Algoritmik = designe og analysere algoritmer**

# Algoritmik

**Algoritmik = designe og analysere algoritmer**

Kvalitet af algoritme:

- **Korrekt** (d.v.s. løser bevisligt problemet).
- Effektiv - lavt **ressourceforbrug**.
  - Tid
  - Plads
- Nem at programmere.
- Problem-specifikke egenskaber.



# Kompleksitet

Kompleksitetsteori  
= studere problemers iboende sværhedsgrad

# Kompleksitet

Kompleksitetsteori  
= studere problemers iboende sværhedsgrad

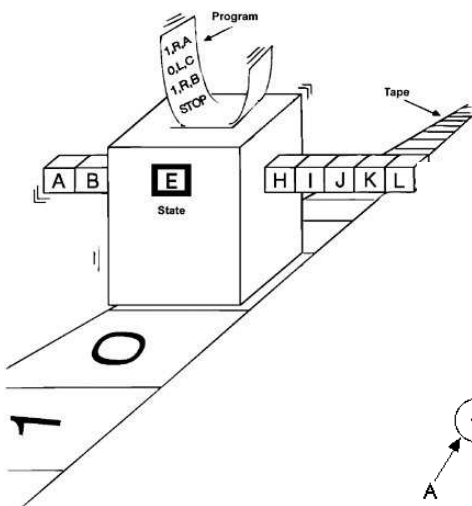
Problemer: Sortering, søgning, korteste veje,...

# Kompleksitet

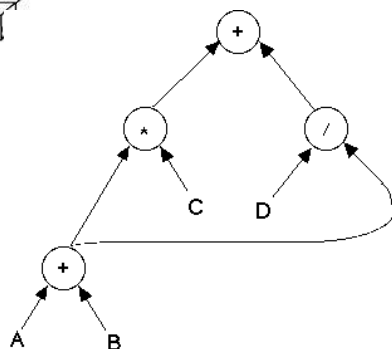
Kompleksitetsteori  
= studere problemers iboende sværhedsgrad

Problemer: Sortering, søgning, korteste veje, ...

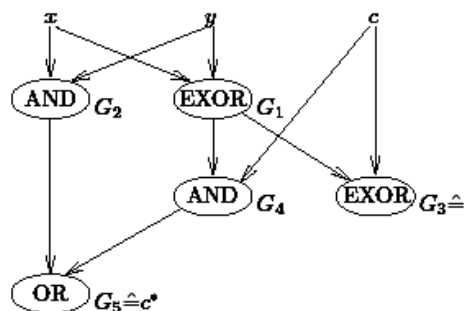
Beregningsmodeller:



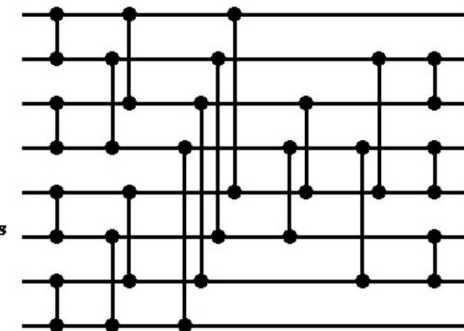
Turing maskine



Aritmetiske netværk



Boolske netværk



Sorterings netværk

# Kompleksitet

Kompleksitetsteori  
= studere problemers iboende sværhedsgrad

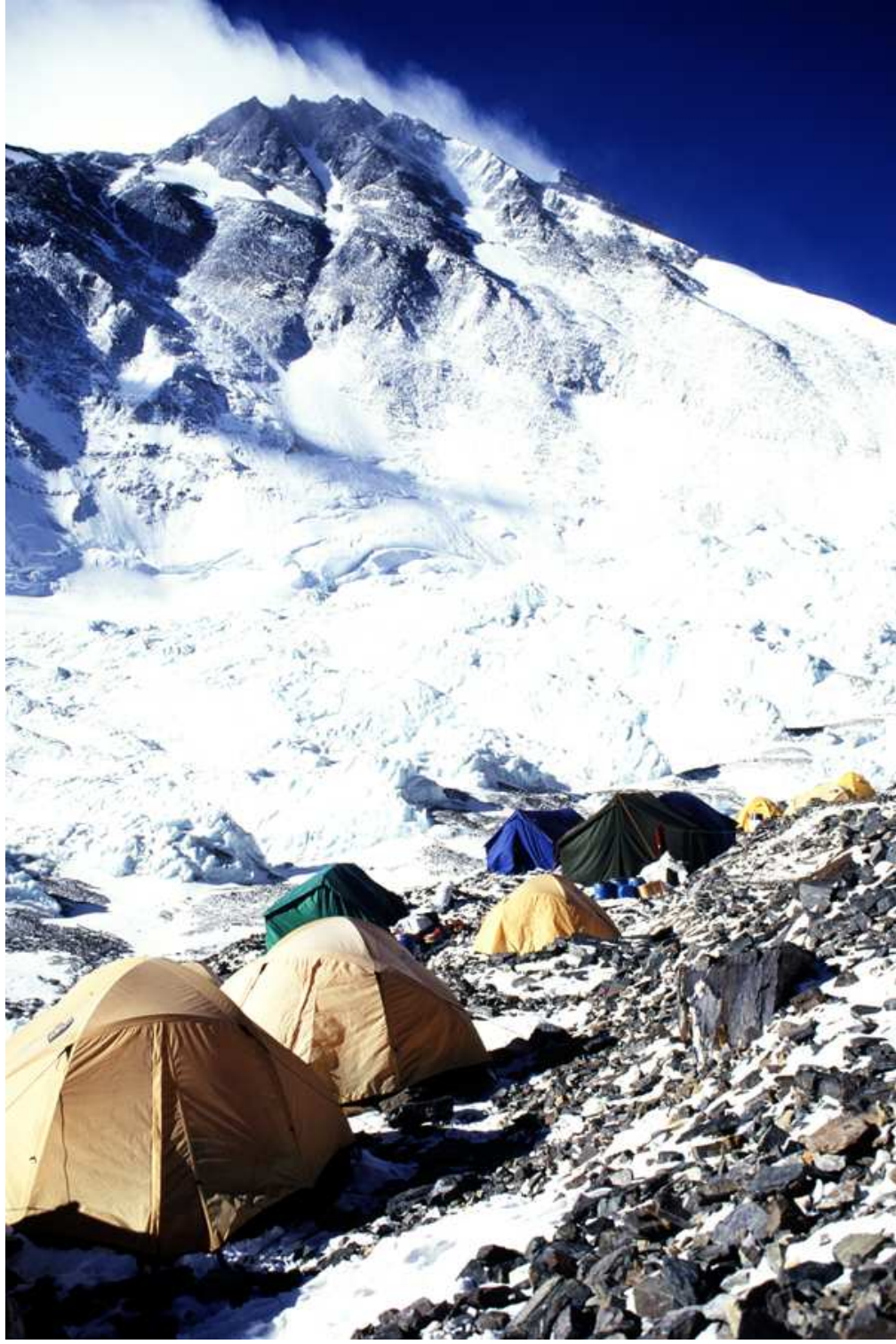
Kompleksitetsklasser:

Klasse( $X, Y$ ) = De problemer, som kan løses i model  $X$  med ressourceforbrug  $Y$ .

Mål:

**Øvre grænser** (d.v.s. algoritmer) og **nedre grænser** (d.v.s. beviser for at **ingen** algoritme i model  $X$  kan løse problemet med ressourceforbrug mindre end  $Y$ ).

# Perspektiverende Datalogikursus



# Format for torsdag

Praktiske opgaver, tænkeopgaver, regneopgaver

# Format for torsdag

Praktiske opgaver, tænkeopgaver, regneopgaver

Program:

- 08.45-09.15 Udlevering af laptops (een person per gruppe).  
og registrering af private maskiner.
- 09.15-12.15 Øvelser.
- 12.15-13.00 Frokost (Storcenter Nord, IT-Parken kantine,  
matematisk kantine, Stakladen, madpakke...)
- 13.00-16.00 Øvelser.

# Format for torsdag

Praktiske opgaver, tænkeopgaver, regneopgaver

Program:

- 08.45-09.15 Udlevering af laptops (een person per gruppe).  
og registrering af private maskiner.
- 09.15-12.15 Øvelser.
- 12.15-13.00 Frokost (Storcenter Nord, IT-Parken kantine,  
matematisk kantine, Stakladen, madpakke...)
- 13.00-16.00 Øvelser.

Eksamen = Tilstedeværelse + Skriftlig Aflevering

Aflevering:

Een udfyldt text-fil. Uploades 15.59 til "BSCW"-systemet.

Lektier: Ingen.



# Format for torsdag

Pointe med øvelser:

- Se eksempler på algoritmiske ideer og metoder.
- Tænke.

Tidstaging (ofte) mindre vigtig end at møde algoritmerne.

Gerne flere i gruppen aktive ad gangen.

Mål:

Motiverende og konkretiserende baggrund for forelæsning fredag.

# Matematik-repetition

Logaritmer (grundtal 2):

$$y = \log_2(x) \Leftrightarrow 2^y = x$$

# Matematik-repetition

Logaritmer (grundtal 2):

$$y = \log_2(x) \Leftrightarrow 2^y = x$$

$x$	1	2	4	8	...	64	...	80	...	128	..
$\log_2(x)$	0	1	2	3		6		6.3219		7	

$$\begin{matrix} \uparrow \\ 2^6 = 64 \end{matrix}$$

# Matematik-repetition

Logaritmer (grundtal 2):

$$y = \log_2(x) \iff 2^y = x$$

$x$	1	2	4	8	...	64	...	80	...	128	..
$\log_2(x)$	0	1	2	3		6		6.3219		7	

$\uparrow$   
 $2^6 = 64$

**NB:**

$$\log(x) = \log_2(x)$$

$$\log(x) \neq \log_{10}(x)$$

$$\log(x) \neq \log_e(x) = \ln(x)$$

(i datalogi)

# Matematik-repetition

$$1 + 2 + 3 + 4 + \dots + N = (N + 1)N/2$$

$$N + N/2 + N/4 + N/8 + \dots + 1 = 2N - 1$$

Ascii-notation for potens:  $2^3$  skrives  $2^3$

# Husk

## Remedier:

Hvert gruppe skal torsdag medbringe: en saks, to-tre ure med sekundvisere, skriveredskaber og lidt kladdepapir, evt. en lommeregner (gerne grafisk)



## Tid og sted:

Torsdag 31. august kl. (8.45) 9:15–16:00 i  
Finlandsgade 24, lokale 157, 159, 164



# Spørgsmål?

I tvivl om noget?



# Spørgsmål?

I tvivl om noget?

Vi ses i morgen!