

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

# Introduktion til DM507

Rolf Fagerberg

Forår 2012

# Mål for i dag

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse  
Værktøjer  
Konkret  
eksempel

Opsummering

## Dagens program:

- 1 Oversigt
- 2 Kursusformat og -formaliteter
- 3 Mål med kursus
- 4 Indhold af kursus
- 5 Algoritmeanalyse
  - Værktøjer
  - Konkret eksempel
- 6 Opsummering

# Kursusformat og -formaliteter

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

**Kursusformat og  
-formaliteter**

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

# Kursusformat og -formaliteter

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

Forudsætninger:

DM502 og DM503

Timer:

50% forelæsninger, 50% øvelser

# Kursusformat og -formaliteter

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse  
Værktøjer

Konkret  
eksempel

Opsummering

## Forudsætninger:

DM502 og DM503

## Timer:

50% forelæsninger, 50% øvelser

## Eksamenform:

### Skriftlig eksamen:

4 timer med bøger, noter, og computer. Karakter efter 7-skala.

### Projekt:

I flere dele. I alt 3 ECTS ud af 10. Karakter B/IB. Skal bestå for at gå til skriftlig eksamen.

# Kursusformat og -formaliteter

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse  
Værktøjer

Konkret  
eksempel

Opsummering

## Forudsætninger:

DM502 og DM503

## Timer:

50% forelæsninger, 50% øvelser

## Eksamenform:

### Skriftlig eksamen:

4 timer med bøger, noter, og computer. Karakter efter 7-skala.

### Projekt:

I flere dele. I alt 3 ECTS ud af 10. Karakter B/IB. Skal bestås for at gå til skriftlig eksamen.

## Materialer:

### Lærebog:

Cormen, Leiserson, Rivest, Stein:  
*Introduction to Algorithms*, 3rd edition, 2009.

Materiale på nettet (gl. eksamenssæt, slides,...)

# Kursusformat og -formaliteter

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

## Forudsætninger:

DM502 og DM503

## Timer:

50% forelæsninger, 50% øvelser

## Eksamenform:

### Skriftlig eksamen:

4 timer med bøger, noter, og computer. Karakter efter 7-skala.

### Projekt:

I flere dele. I alt 3 ECTS ud af 10. Karakter B/IB. Skal bestås for at gå til skriftlig eksamen.

## Materialer:

### Lærebog:

Cormen, Leiserson, Rivest, Stein:  
*Introduction to Algorithms*, 3rd edition, 2009.

Materiale på nettet (gl. eksamenssæt, slides,...)

# Perspektivering: Kursets placering i det store billede

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

**Mål med kursus**

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering



# Perspektivering: Kursets placering i det store billede

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

**Mål med kursus**

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

Generelt mål i datalogi: **Få computer til at udføre en opgave.**

# Perspektivering: Kursets placering i det store billede

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

**Mål med kursus**

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

Generelt mål i datalogi: **Få computer til at udføre en opgave.**

Relaterede spørgsmål:

# Perspektivering: Kursets placering i det store billede

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

Generelt mål i datalogi: **Få computer til at udføre en opgave.**

Relaterede spørgsmål:

- **Hvordan skrives programmer?**

(Java-)programmering, programmeringssprog, software engineering (software arkitektur, udviklingsmetodikker, projektstyring, usability) [DM502, DM503, DM509, DM529, DM530, DM531].

# Perspektivering: Kursets placering i det store billede

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

Generelt mål i datalogi: **Få computer til at udføre en opgave.**

Relaterede spørgsmål:

- **Hvordan skrives programmer?**

(Java-)programmering, programmeringssprog, software engineering (software arkitektur, udviklingsmetodikker, projektstyring, usability) [DM502, DM503, DM509, DM529, DM530, DM531].

- **Hvordan skal programmet løse opgaven?**

Algoritmer og datastrukturer [DM507, DM508, DM515, MM518]

# Perspektivering: Kursets placering i det store billede

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

Generelt mål i datalogi: **Få computer til at udføre en opgave.**

Relaterede spørgsmål:

- **Hvordan skrives programmer?**  
(Java-)programmering, programmeringssprog, software engineering (software arkitektur, udviklingsmetodikker, projektstyring, usability) [DM502, DM503, DM509, DM529, DM530, DM531].
- **Hvordan skal programmet løse opgaven?**  
Algoritmer og datastrukturer [DM507, DM508, DM515, MM518]
- **(Hvor godt) er det overhovedet muligt at løse opgaven?**  
Nedre grænser, kompleksitet, beregnelighed [DM508, DM517].

# Perspektivering: Kursets placering i det store billede

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

Generelt mål i datalogi: **Få computer til at udføre en opgave.**

Relaterede spørgsmål:

- **Hvordan skrives programmer?**  
(Java-)programmering, programmeringssprog, software engineering (software arkitektur, udviklingsmetodikker, projektstyring, usability) [DM502, DM503, DM509, DM529, DM530, DM531].
- **Hvordan skal programmet løse opgaven?**  
Algoritmer og datastrukturer [DM507, DM508, DM515, MM518]
- **(Hvor godt) er det overhovedet muligt at løse opgaven?**  
Nedre grænser, kompleksitet, beregnelighed [DM508, DM517].
- **Hvordan fungerer maskinen der udfører opgaven?**  
Baggrundviden om computerarkitektur og operativsystemer [DM506, DM510].

# Perspektivering: Kursets placering i det store billede

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

Generelt mål i datalogi: **Få computer til at udføre en opgave.**

Relaterede spørgsmål:

- **Hvordan skrives programmer?**  
(Java-)programmering, programmeringssprog, software engineering (software arkitektur, udviklingsmetodikker, projektstyring, usability) [DM502, DM503, DM509, DM529, DM530, DM531].
- **Hvordan skal programmet løse opgaven?** ⇐  
**Algoritmer og datastrukturer** [DM507, DM508, DM515, MM518]
- **(Hvor godt) er det overhovedet muligt at løse opgaven?**  
Nedre grænser, kompleksitet, beregnelighed [DM508, DM517].
- **Hvordan fungerer maskinen der udfører opgaven?**  
Baggrundsviden om computerarkitektur og operativsystemer [DM506, DM510].

# Fokus: *Hvordan* skal programmet løse opgaven?

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

**Mål med kursus**

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering



# Fokus: *Hvordan* skal programmet løse opgaven?

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

Relevante opgaver for ethvert beregningsproblem:

- 1 **Find** (mindst) **een** algoritme der løser problemet.

# Fokus: *Hvordan* skal programmet løse opgaven?

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

Relevante opgaver for ethvert beregningsproblem:

- 1 **Find** (mindst) **een** algoritme der løser problemet.
- 2 **Sammenlign flere** algoritmer der løser problemet.

# Fokus: *Hvordan* skal programmet løse opgaven?

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

Relevante opgaver for ethvert beregningsproblem:

- 1 **Find** (mindst) **een** algoritme der løser problemet.
- 2 **Sammenlign flere** algoritmer der løser problemet.
- 3 Hvad er den **bedste** algoritme der kan findes?

# Fokus: *Hvordan* skal programmet løse opgaven?

Relevante opgaver for ethvert beregningsproblem:

- 1 **Find** (mindst) **een** algoritme der løser problemet.
- 2 **Sammenlign flere** algoritmer der løser problemet.
- 3 Hvad er den **bedste** algoritme der kan findes?

**Algoritme = løsningsmetode.**

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

# Fokus: *Hvordan* skal programmet løse opgaven?

Relevante opgaver for ethvert beregningsproblem:

- 1 **Find** (mindst) **een** algoritme der løser problemet.
- 2 **Sammenlign flere** algoritmer der løser problemet.
- 3 Hvad er den **bedste** algoritme der kan findes?

**Algoritme = løsningsmetode.**

Tilpas præcist skrevet ned: præcis tekst, pseudo-kode, flow-diagrammer, formler, . . .

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

# Fokus: *Hvordan* skal programmet løse opgaven?

Relevante opgaver for ethvert beregningsproblem:

- 1 **Find** (mindst) **een** algoritme der løser problemet.
- 2 **Sammenlign flere** algoritmer der løser problemet.
- 3 Hvad er den **bedste** algoritme der kan findes?

**Algoritme = løsningsmetode.**

Tilpas præcist skrevet ned: præcis tekst, pseudo-kode, flow-diagrammer, formler, . . .

**Datastruktur = data + effektive operationer herpå.**

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

# Fokus: *Hvordan* skal programmet løse opgaven?

Relevante opgaver for ethvert beregningsproblem:

- 1 **Find** (mindst) **een** algoritme der løser problemet.
- 2 **Sammenlign flere** algoritmer der løser problemet.
- 3 Hvad er den **bedste** algoritme der kan findes?

**Algoritme = løsningsmetode.**

Tilpas præcist skrevet ned: præcis tekst, pseudo-kode, flow-diagrammer, formler, . . .

**Datastruktur = data + effektive operationer herpå.**

Forskellige datastrukturer gemmer forskellige typer data og/eller tilbyder forskellige operationer. Har stor anvendelse som delelement i algoritmer.

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

# Udvikling og vurdering af algoritmer

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

- 1 Find (mindst) **een** algoritme der løser problemet.
- 2 **Sammenlign flere** algoritmer der løser problemet.
- 3 Hvad er den **bedste** algoritme der kan findes?



# Udvikling og vurdering af algoritmer

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

- 1 Find (mindst) **een** algoritme der løser problemet.
- 2 **Sammenlign flere** algoritmer der løser problemet.
- 3 Hvad er den **bedste** algoritme der kan findes?

**Punkt 1:** Kræver ideer, tænkearbejde, erfaring, og et bagkatalog af algoritmer. Korrekthed: ved analyse eller implementation/afprøvning?

# Udvikling og vurdering af algoritmer

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

- 1 **Find** (mindst) **een** algoritme der løser problemet.
- 2 **Sammenlign flere** algoritmer der løser problemet.
- 3 Hvad er den **bedste** algoritme der kan findes?

**Punkt 1:** Kræver ideer, tænkearbejde, erfaring, og et bagkatalog af algoritmer. Korrekthed: ved analyse eller implementation/afprøvning?

**Punkt 2:** Kræver definition af hvad er kvalitet. Sammenligning: ved analyse eller implementation/afprøvning?

# Udvikling og vurdering af algoritmer

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

- 1 **Find** (mindst) **een** algoritme der løser problemet.
- 2 **Sammenlign flere** algoritmer der løser problemet.
- 3 Hvad er den **bedste** algoritme der kan findes?

**Punkt 1:** Kræver ideer, tænkearbejde, erfaring, og et bagkatalog af algoritmer. Korrekthed: ved analyse eller implementation/afprøvning?

**Punkt 2:** Kræver definition af hvad er kvalitet. Sammenligning: ved analyse eller implementation/afprøvning?

**Analyse** (DM507): Giver (høj) sikkerhed for korrekthed. Sparer arbejde. Sammenligning upåvirket af: maskine, sprog, programmør, konkrete input.

# Udvikling og vurdering af algoritmer

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

- 1 **Find** (mindst) **een** algoritme der løser problemet.
- 2 **Sammenlign flere** algoritmer der løser problemet.
- 3 Hvad er den **bedste** algoritme der kan findes?

**Punkt 1:** Kræver ideer, tænkearbejde, erfaring, og et bagkatalog af algoritmer. Korrekthed: ved analyse eller implementation/afprøvning?

**Punkt 2:** Kræver definition af hvad er kvalitet. Sammenligning: ved analyse eller implementation/afprøvning?

**Analyse** (DM507): Giver (høj) sikkerhed for korrekthed. Sparer arbejde. Sammenligning upåvirket af: maskine, sprog, programmør, konkrete input. (**Men:** relevans for virkelighed skal holdes for øje.)

# Udvikling og vurdering af algoritmer

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

- 1 **Find** (mindst) **een** algoritme der løser problemet.
- 2 **Sammenlign flere** algoritmer der løser problemet.
- 3 Hvad er den **bedste** algoritme der kan findes?

**Punkt 1:** Kræver ideer, tænkearbejde, erfaring, og et bagkatalog af algoritmer. Korrekthed: ved analyse eller implementation/afprøvning?

**Punkt 2:** Kræver definition af hvad er kvalitet. Sammenligning: ved analyse eller implementation/afprøvning?

**Analyse** (DM507): Giver (høj) sikkerhed for korrekthed. Sparer arbejde. Sammenligning upåvirket af: maskine, sprog, programmør, konkrete input. (**Men:** relevans for virkelighed skal holdes for øje.)

I alle byggefag analyserer og planlægger man før man bygger (tænk storebæltsbro). Din fremtidige chef vil (bør) forlange det!

# Udvikling og vurdering af algoritmer

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

- 1 **Find** (mindst) **een** algoritme der løser problemet.
- 2 **Sammenlign flere** algoritmer der løser problemet.
- 3 Hvad er den **bedste** algoritme der kan findes?

**Punkt 1:** Kræver ideer, tænkearbejde, erfaring, og et bagkatalog af algoritmer. Korrekthed: ved analyse eller implementation/afprøvning?

**Punkt 2:** Kræver definition af hvad er kvalitet. Sammenligning: ved analyse eller implementation/afprøvning?

**Analyse** (DM507): Giver (høj) sikkerhed for korrekthed. Sparer arbejde. Sammenligning upåvirket af: maskine, sprog, programmør, konkrete input. (**Men:** relevans for virkelighed skal holdes for øje.)

I alle byggefag analyserer og planlægger man før man bygger (tænk storebæltsbro). Din fremtidige chef vil (bør) forlange det!

Og bemærk: **Punkt 3** kan *kun* afklares med analyse.

# Målsætning for kurset

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

**Mål med kursus**

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

# Målsætning for kurset

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

**Mål med kursus**

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

DM507 er en værktøjskasse af algoritmer for fundamentale problemer, samt metoder til at udvikle og analysere nye algoritmer og varianter af eksisterende.



# Målsætning for kurset

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

DM507 er en værktøjskasse af algoritmer for fundamentale problemer, samt metoder til at udvikle og analysere nye algoritmer og varianter af eksisterende.



# Målsætning for kurset

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

DM507 er en værktøjskasse af algoritmer for fundamentale problemer, samt metoder til at udvikle og analysere nye algoritmer og varianter af eksisterende.



# Målsætning for kurset

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

Regneøvelser og programmeringsprojekter træner brugen af værktøjskassen i praksis.



# Målsætning for kurset

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

Regneøvelser og programmeringsprojekter træner brugen af værktøjskassen i praksis.



Undervejs begejstres man også over mødet med smukke og smarte ideer brugt i algoritmer og deres analyse.



# Formel målbeskrivelse for eksamen.

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

**Mål med kursus**

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

# Formel målbeskrivelse for eksamen.

Ved kursets afslutning forventes den studerende at kunne:

- Anvende algoritmerne fra kurset på konkrete problemer.
- Argumentere præcist for en algoritmes korrekthed eller mangel på samme.
- Bestemme en algoritmes asymptotiske køretid.
- Tilpasse kendte algoritmer og datastrukturer til specialtilfælde af kendte problemer og til nye problemer.
- Designe nye algoritmer til at løse problemer, som i natur minder om problemer fra kurset. Herunder give en præcis beskrivelse af algoritmen, feks. vha. pseudokode.
- Foretage fornuftige valg af datastruktur.
- Designe nye datastrukturer baseret på kendte datastrukturer.
- Designe og implementere et større program, som bl.a. anvender algoritmer og datastrukturer fra kurset.
- Argumentere præcist for de valg, der foretages i forbindelse med foregående 4 punkter.

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

# Indhold af kursus

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

**Indhold af kursus**

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

# Indhold af kursus

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

## Algoritmer:

- Korrekthed og kompleksitetsanalyse
- Del og hersk algoritmer
- Grådige algoritmer
- Dynamisk programmering
- Sortering
- Graf-algoritmer
- Huffman-kodning

## Datastrukturer:

- Træer
- Ordbøger
- Prioritetskøer
- Disjunkte mængder



# I DM503 (DM526) har I allerede mødt:

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

**Indhold af kursus**

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

# I DM503 (DM526) har I allerede mødt:

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse  
Værktøjer

Konkret  
eksempel

Opsummering

## Algoritmer:

- Binær søgning
- Sortering: SelectionSort, (InsertionSort, Mergesort, (Quicksort)).

## Datastrukturer:

- Kø, stak, liste.
- Binære søgetræer, ubalancerede.

# Mere om algoritmeanalyse

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

**Algoritmeanalyse**

Værktøjer

Konkret  
eksempel

Opsummering

# Mere om algoritmeanalyse

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

**Algoritmeanalyse**

Værktøjer

Konkret  
eksempel

Opsummering

Korrekthed af algoritmer:

- Terminerer (for alle input).
- Korrekt output (for alle input) når terminerer.

# Mere om algoritmeanalyse

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

**Algoritmeanalyse**

Værktøjer

Konkret  
eksempel

Opsummering

Korrekthed af algoritmer:

- Terminerer (for alle input).
- Korrekt output (for alle input) når terminerer.

Kvalitet af algoritmer:

- Hastighed
- Pladsforbrug
- Komplexitet af implementation
- Ekstra egenskaber (problemspecifikke), f.eks. stabilitet af sortering.

# Mere om algoritmeanalyse

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

**Algoritmeanalyse**

Værktøjer

Konkret  
eksempel

Opsummering

Korrekthed af algoritmer:

- Terminerer (for alle input).
- Korrekt output (for alle input) når terminerer.

Kvalitet af algoritmer:

- Hastighed
- Pladsforbrug
- Komplexitet af implementation
- Ekstra egenskaber (problemspecifikke), f.eks. stabilitet af sortering.

For dette kræves modellering af problem og maskine, en definition af kvalitet, samt en matematisk/datalogisk værktøjskasse af analyseredskaber.

# Værktøjer til algoritmeanalyse

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

**Værktøjer**

Konkret  
eksempel

Opsummering

# Værktøjer til algoritmeanalyse

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

**Værktøjer**

Konkret  
eksempel

Opsummering

- Model af problem. Problemspecifikt.



# Værktøjer til algoritmeanalyse

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

**Værktøjer**

Konkret  
eksempel

Opsummering

- Model af problem. Problemspecifikt.
- Model af maskine. Ofte RAM-modellen (alias von Neumann modellen).

# Værktøjer til algoritmeanalyse

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

**Værktøjer**

Konkret  
eksempel

Opsummering

- Model af problem. Problemspecifikt.
- Model af maskine. Ofte RAM-modellen (alias von Neumann modellen).
- Mål for ressourceforbrug (tid og plads).

# Værktøjer til algoritmeanalyse

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

**Værktøjer**

Konkret  
eksempel

Opsummering

- Model af problem. Problemspecifikt.
- Model af maskine. Ofte RAM-modellen (alias von Neumann modellen).
- Mål for ressourceforbrug (tid og plads).
- Matematiske/datalogiske analyseværktøjer: Løkkeinvarianter, induktion, rekursionsligninger, potentialfunktioner (til amortiseret analyse [DM508]),...

# RAM-modellen

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

**Værktøjer**

Konkret  
eksempel

Opsummering

# RAM-modellen

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

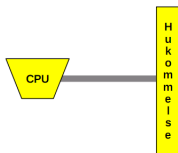
Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering



- En CPU
- En hukommelse ( $\sim$  uendeligt array af celler).
- Et antal basale operationer: add, sub, mult, shift, sammenligning, flyt dataelement mellem celler i hukommelsen. Disse antages alle at tage samme tid.
- Tid for en algoritme sættes lig antal basale operationer udført.
- Plads for en algoritme sættes lig det maksimale antal hukommelsesceller som er optaget under kørslen.

# Mål for ressourceforbrug: Voksehastighed

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

**Værktøjer**

Konkret  
eksempel

Opsummering

# Mål for ressourceforbrug: Voksehastighed

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

**Værktøjer**

Konkret  
eksempel

Opsummering

For alle problemer må det antages at større input giver større ressourceforbrug (tid, plads). Forbruget kan derfor ses som en funktion  $f(n)$  af inputstørrelsen  $n$ .

# Mål for ressourceforbrug: Voksehastighed

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

For alle problemer må det antages at større input giver større ressourceforbrug (tid, plads). Forbruget kan derfor ses som en funktion  $f(n)$  af inputstørrelsen  $n$ .

(Her skal der for hvert problem vedtages et mål for inputstørrelse – f.eks. antal dataelementer eller bytes i input.)



# Mål for ressourceforbrug: Voksehastighed

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse  
Værktøjer

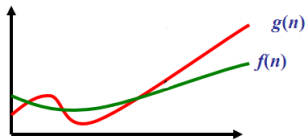
Konkret  
eksempel

Opsummering

For alle problemer må det antages at større input giver større ressourceforbrug (tid, plads). Forbruget kan derfor ses som en funktion  $f(n)$  af inputstørrelsen  $n$ .

(Her skal der for for hvert problem vedtages et mål for inputstørrelse – f.eks. antal dataelementer eller bytes i input.)

Vi har derfor brug for at sammenligne funktioner. Det relevante mål er **voksehastighed** - en hurtigere voksende funktion vil altid overhale en langsomt voksende funktion når  $n$  bliver stor nok. Og for små  $n$  er (næsten) alle algoritmer hurtige.



# Mål for ressourceforbrug: Voksehastighed

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

Eksempler (stigende voksehastighed):

$$1, \log n, \sqrt{n}, n/\log n, n, n \log n, n\sqrt{n}, n^2, n^3, n^{10}, 2^n$$

# Mål for ressourceforbrug: Voksehastighed

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

Eksempler (stigende voksehastighed):

$$1, \log n, \sqrt{n}, n/\log n, n, n \log n, n\sqrt{n}, n^2, n^3, n^{10}, 2^n$$

Mere præcis definition af begrebet [asymptotisk voksehastighed](#) næste gang.

# Worst case kompleksitet

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

**Værktøjer**

Konkret  
eksempel

Opsummering

# Worst case kompleksitet

For en givet størrelse  $n$  af input er der ofte mange forskellige input instanser. Algoritmen har som regel forskelligt ressourceforbrug på hver af disse. Hvilket skal vi bruge til at vurdere ressourceforbruget?

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

**Værktøjer**

Konkret  
eksempel

Opsummering

# Worst case kompleksitet

For en givet størrelse  $n$  af input er der ofte mange forskellige input instanser. Algoritmen har som regel forskelligt ressourceforbrug på hver af disse. Hvilket skal vi bruge til at vurdere ressourceforbruget?

- 1 Worst case (max over alle input af størrelse  $n$ )
- 2 Average case (gennemsnit over en fordeling af input af størrelse  $n$ )
- 3 Best case (min over alle input af størrelse  $n$ )

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

# Worst case kompleksitet

For en givet størrelse  $n$  af input er der ofte mange forskellige input instanser. Algoritmen har som regel forskelligt ressourceforbrug på hver af disse. Hvilket skal vi bruge til at vurdere ressourceforbruget?

- 1 Worst case (max over alle input af størrelse  $n$ )
- 2 Average case (gennemsnit over en fordeling af input af størrelse  $n$ )
- 3 Best case (min over alle input af størrelse  $n$ )

Worst case giver garanti. Ofte repræsentativ for average case (men nogen gange betydeligt mere pessimistisk).

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse  
Værktøjer

Konkret  
eksempel

Opsummering

# Worst case kompleksitet

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse  
Værktøjer

Konkret  
eksempel

Opsummering

For en givet størrelse  $n$  af input er der ofte mange forskellige input instanser. Algoritmen har som regel forskelligt ressourceforbrug på hver af disse. Hvilket skal vi bruge til at vurdere ressourceforbruget?

- 1 Worst case (max over alle input af størrelse  $n$ )
- 2 Average case (gennemsnit over en fordeling af input af størrelse  $n$ )
- 3 Best case (min over alle input af størrelse  $n$ )

Worst case giver garanti. Ofte repræsentativ for average case (men nogen gange betydeligt mere pessimistisk).

Average case: Hvilken fordeling? Er den realistisk? Ofte svær analyse at gennemføre (matematisk svær)



# Worst case kompleksitet

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse  
Værktøjer

Konkret  
eksempel

Opsummering

For en givet størrelse  $n$  af input er der ofte mange forskellige input instanser. Algoritmen har som regel forskelligt ressourceforbrug på hver af disse. Hvilket skal vi bruge til at vurdere ressourceforbruget?

- 1 Worst case (max over alle input af størrelse  $n$ )
- 2 Average case (gennemsnit over en fordeling af input af størrelse  $n$ )
- 3 Best case (min over alle input af størrelse  $n$ )

Worst case giver garanti. Ofte repræsentativ for average case (men nogen gange betydeligt mere pessimistisk).

Average case: Hvilken fordeling? Er den realistisk? Ofte svær analyse at gennemføre (matematisk svær)

Best case: Giver som regel ikke megen relevant information.

# Worst case kompleksitet

For en givet størrelse  $n$  af input er der ofte mange forskellige input instanser. Algoritmen har som regel forskelligt ressourceforbrug på hver af disse. Hvilket skal vi bruge til at vurdere ressourceforbruget?

- 1 Worst case (max over alle input af størrelse  $n$ )
- 2 Average case (gennemsnit over en fordeling af input af størrelse  $n$ )
- 3 Best case (min over alle input af størrelse  $n$ )

Worst case giver garanti. Ofte repræsentativ for average case (men nogen gange betydeligt mere pessimistisk).

Average case: Hvilken fordeling? Er den realistisk? Ofte svær analyse at gennemføre (matematisk svær)

Best case: Giver som regel ikke megen relevant information.

Næsten alle analyser i dette kursus er worst case.

# Konkret eksempel på algoritmeanalyse

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

**Konkret  
eksempel**

Opsummering

Jul i Valhalla...

# Hvad har vi lært i dag?

Introduktion til  
DM507

Rolf Fagerberg

Oversigt

Kursusformat og  
-formaliteter

Mål med kursus

Indhold af kursus

Algoritmeanalyse

Værktøjer

Konkret  
eksempel

Opsummering

- 1 Oversigt
- 2 Kursusformat og -formaliteter
- 3 Mål med kursus
- 4 Indhold af kursus
- 5 Algoritmeanalyse
  - Værktøjer
  - Konkret eksempel
- 6 Opsummering