

Algoritmer og datastrukturer

Introduktion til kurset

Rolf Fagerberg

Forår 2026

Hvem er vi?

Underviser:

- ▶ Rolf Fagerberg, Institut for Matematik og Datalogi (IMADA)
Forskningsområde: algoritmer og datastrukturer

Deltagere:

- ▶ BA i Datalogi (2. sem.)
- ▶ BA i Kunstig Intelligens (2. sem.)
- ▶ BA i Software Engineering (4. sem.)
- ▶ BA i Matematik-Økonomi (4. sem.)
- ▶ BA i Anvendt Matematik (4. sem.)
- ▶ BA sidefag i Datalogi (6. sem.)
- ▶ KA profil i Data Science (2./8. sem.)

Bemærk stor diversitet: forskellige semestre i uddannelsen, forskellige mængder af programmering og af matematiske fag på uddannelsen.

Tre (fire) kurser i ét

- DM578:
 - ▶ Algoritmer og datastrukturer (7.5 ETCS)
 - ▶ Skriftlig MC eksamen (3 timer)
- DM507/DS814:
 - ▶ Algoritmer og datastrukturer (7.5 ETCS)
 - ▶ Programmeringsprojekt i tre dele (2.5 ETCS)
 - ▶ Skriftlig MC eksamen (3 timer)
- SE4-DMAD:
 - ▶ Algoritmer og datastrukturer (7.5 ETCS)
 - ▶ Diskret matematik (2.5 ETCS)
 - ▶ Skriftlig MC eksamen (3 timer og 3 kvarter)

Diskret matematik har separate forelæsninger (Lene Monrad Favrholdt) og øvelsestimer. Fredage 10-12 i ti uger.

Der er et ITS-kursusrum for DM507/DM578/DS814 og et andet ITS-kursusrum for SE4-DMAD.

Kursets format (DM578)

Forudsætninger:

Programmering i Python eller Java, lidt matematisk modenhed

Format:

Forelæsninger (f-timer) ved Rolf Fagerberg.

Opgaveregning (e-timer) ved instruktør.

Arbejde selv og i studiegrupper.

Eksamenform:

Skriftlig eksamen (juni), 7.5 ECTS:

Multiple-choice. Kun med bøger og noter (ingen programmer eller internet). Karakter efter 7-skala. Mål: check af kendskab til stoffet. [NB: reeksamen er *mundtlig*.]

Kursets format (DM507/DS814)

Forudsætninger:

Programmering i Python, lidt matematisk modenhed

Format:

Forelæsninger (f-timer) ved Rolf Fagerberg.

Opgaveregning (e-timer) ved Instruktør.

Arbejde selv og i studiegrupper.

Eksamenform:

Skriftlig eksamen (juni), 7.5 ECTS:

Multiple-choice. Kun med bøger og noter (ingen programmer eller internet). Karakter efter 7-skala. Mål: check af kendskab til stoffet. [NB: reeksamen er *mundtlig*.]

Projekt undervejs, 2.5 ECTS, Python:

I tre dele. Karakter B/IB. Skal ikke bestås for at gå til skriftlig eksamen. Mål: træne overførsel af stoffet til praksis (programmering).

Kursets format (SE4-DMAD)

Forudsætninger:

Programmering i C# eller Python, lidt matematisk modenhed

Format:

Forelæsninger (f-timer) ved Rolf Fagerberg og Lene Monrad Favrholt.

Opgaveregning (e-timer) ved instruktør.

Arbejde selv og i studiegrupper.

Eksamenform:

Skriftlig eksamen (juni), 10 ECTS:

Multiple-choice. Kun med bøger og noter (ingen programmer eller internet). Karakter efter 7-skala. Mål: check af kendskab til stoffet. [NB: reeksamen er *mundtlig*.]

Materialer i algoritmer og datastrukturer

Lærebog:

Cormen, Leiserson, Rivest, Stein:
Introduction to Algorithms, 4th edition, 2022.

(Jeg vil også angive læsestof og opgaver ud fra 3rd edition.)

Andet læremateriale på kursets webside:

Slides fra forelæsninger
Opgaver til øvelsestimer
Tidligere eksamenssæt
Projektet

Forventet arbejdsindsats (DM578)

- ▶ Skim stof før forelæsning: 0,5 timer \Leftarrow mindst vigtig
- ▶ Forelæsning: 2 timer
- ▶ Læs stof efter forelæsning: 1,5 timer
- ▶ Opgaveregning (hjemme): 3 timer \Leftarrow mest vigtig
- ▶ Opgaveregning (klasse): 2 timer \Leftarrow mest vigtig

Ovenstående i gennemsnit 1.5 gang per uge over 14 uger. Dertil følgende én gang:

- ▶ Eksamenslæsning: 40 timer
- ▶ Eksamen: 3 timer

I alt: $14 \cdot 1.5 \cdot 9 + 40 + 3 = 232$ timer

$7.5 \text{ ECTS} = 1/8 \text{ årsværk} = 1650/8 \text{ timer} = 206 \text{ timer}$

Forventet arbejdsindsats (DM507/DS814)

- ▶ Skim stof før forelæsning: 0,5 timer \Leftarrow mindst vigtig
- ▶ Forelæsning: 2 timer
- ▶ Læs stof efter forelæsning: 1,5 timer
- ▶ Opgaveregning (hjemme): 3 timer \Leftarrow mest vigtig
- ▶ Opgaveregning (klasse): 2 timer \Leftarrow mest vigtig

Ovenstående i gennemsnit 1.5 gang per uge over 14 uger. Dertil følgende én gang:

- ▶ Projektet: 15+15+25 timer
- ▶ Eksamenslæsning: 40 timer
- ▶ Eksamen: 3 timer

I alt: $14 \cdot 1.5 \cdot 9 + 55 + 40 + 3 = 287$ timer

10 ECTS = $1/6$ årsværk = $1650/6$ timer = 275 timer

Forventet arbejdsindsats (SE4-DMAD)

- ▶ Skim stof før forelæsning: 0,5 timer \Leftarrow mindst vigtig
- ▶ Forelæsning: 2 timer
- ▶ Læs stof efter forelæsning: 1,5 timer
- ▶ Opgaveregning (hjemme): 3 timer \Leftarrow mest vigtig
- ▶ Opgaveregning (klasse): 2 timer \Leftarrow mest vigtig

Ovenstående i gennemsnit 1.5 per uge over 14 uger (Rolf) og 0.5 gang per uge over 10 uger (Lene). Dertil følgende én gang:

- ▶ Eksamenslæsning: 40 timer
- ▶ Eksamen: 4 timer

I alt: $(14 \cdot 1.5 + 10 \cdot 0.5) \cdot 9 + 40 + 4 = 278$ timer

10 ECTS = $1/6$ årsværk = $1650/6$ timer = 275 timer

Kursets formål og plads i det store billede

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

Relaterede spørgsmål:

- ▶ **Hvordan skrives programmer?**
Programmering, programmeringssprog, software engineering.
- ▶ **Hvordan skal programmet løse opgaven?** \Leftarrow DM507/DM578/...
Algoritmer og datastrukturer, databasesystemer, lineær algebra med anvendelser, data mining og machine learning.
- ▶ **(Hvor godt) er det overhovedet muligt at løse opgaven?**
Nedre grænser, kompleksitet, beregnelighed.
- ▶ **Hvordan fungerer maskinen der udfører opgaven?**
Baggrundsviden om computerarkitektur og operativsystemer.

Hvordan skal programmet løse opgaven?

Algoritme = løsningsmetode.

Tilpas præcist skrevet ned: tegning, tekst, pseudo-kode,...

Datastruktur = data + effektive operationer herpå.

Forskellige niveauer af løsning:

1. **Opfind** én algoritme som løser opgaven.
2. **Sammenlign** flere algoritmer som løser opgaven.
3. Hvad er den **bedst mulige** algoritme som løser opgaven?

Udvikling og vurdering af algoritmer

1. **Opfind** en algoritme som løser opgaven: Kræver ideer, erfaring, og en værktøjskasse med både eksisterende algoritmer og metoder til at udvikle nye.
2. **Sammenlign** flere algoritmer som løser opgaven: Kræver definition af hvad kvalitet er (ofte: kvalitet = lavt tidsforbrug).
3. Hvad er den **bedst mulige** algoritme som løser opgaven?

Analyse (tænkearbejde, argumenter, beviser): godt værktøj til punkt 1, 2 og 3. Giver høj sikkerhed for korrekthed af metoden/idéen. Sparer implementationsarbejde. Sammenligning upåvirket af: maskine, sprog, programmør, og valg af testdata.

Afprøvning (implementation, test): godt værktøj til punkt 1 og 2. Kan udforske ideer, fange implementationsfejl, belyse ting som ikke fanges af analysen.

Udvikling og vurdering af algoritmer

DM507/DM578/DS814/SE4-DMAD vil have **mest fokus på analyse**, lidt mindre på implementation og afprøvning.

I alle byggefag analyserer og planlægger man, før man bygger (tænk f.eks. storebæltsbro).

Hvad med GenAI?

GenAI er en måde **hurtigt** at skaffe sig kode, som **måske** virker.

Et universitetstudie bør klæde jer på, så I kan **vurdere** output af GenAI.
Derfor bliver I nødt til at **forstå** principperne selv.

Forståelse kommer af **hjernens arbejde** med stoffet. Ikke øjnenes.

Målsætning for kurset

DM507 giver dig en **værktøjskasse af algoritmer** for fundamentale opgaver, samt **metoder til at udvikle og analysere nye algoritmer** og varianter af eksisterende.



Målsætning for kurset

Øvelser og programmeringsprojekter øger din **forståelse** for værktøjerne og træner dig i brugen af dem.



Undervejs begejstres du måske også over **smarte og elegante ideer** i algoritmer og analyser.



Konkret indhold af kurset

Algoritmer:

- ▶ Analyse af algoritmer: korrekthed og køretid (analyseværktøj)
- ▶ Del og hersk algoritmer (algoritmemetode)
- ▶ Grådige algoritmer (algoritmemetode)
- ▶ Dynamisk programmering (algoritmemetode)
- ▶ Konkrete algoritmer for sortering, graf-problemer (BFS, DFS, korteste veje, udspændende træer, ...), fil-komprimering, multiplikation af matricer, ...

Datastrukturer:

- ▶ Ordbøger (søgetræer og hashing)
- ▶ Prioritetskøer (heaps)
- ▶ Disjunkte mængder