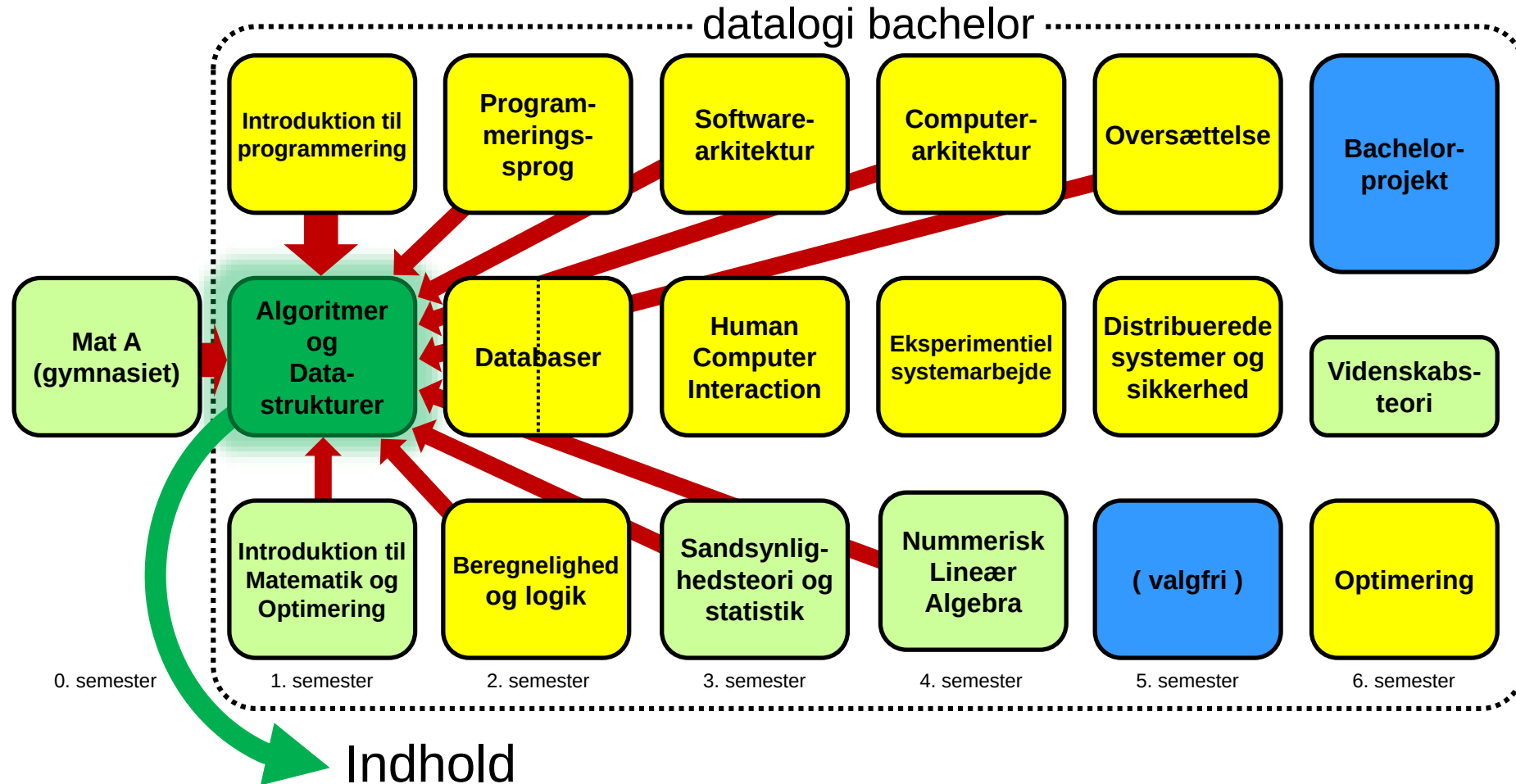


Algoritmer og Datastrukturer

Gerth Stølting Brodal
gerth@cs.au.dk

➔ faglige forudsætninger
(ideelle verden)

Eksamen : januar
Reeksamen : maj



- **Introduktion til algoritmer og datastrukturer**
 - Eksempler på anvendelse
 - Avancerede algoritmer og datastrukturer
 - Forskningsresultater
- ambitiøst ↓

Kursuselementer

Hvis du er forhindret i at møde fysisk på campus grundet coronavirus, kan du søge om at få en studieassistent (studerende.au.dk/studieassistent/)

Undervisningsformer

Forelæsninger 2 + 2 timer/uge [fysisk + streames på Zoom + efterfølgende video]

Teoretiske øvelser "TØ" 3 timer/uge [fysisk]

Studiecafé 5 x 2 timer/uge [Zoom]

Online diskussionsforum [Blackboard]

Forelæsningerne gennemgår stoffet
Til øvelserne *arbejder* man med stoffet

Hjælp til ugeopgaver og afleveringer af en instruktør (cs.au.dk/studiecafe)

Afleveringer (grupper 1-3 personer)

9 teoretiske afleveringer (skal godkendes)

4 ugers programmeringsopgaver (del af karakteren)

Træner algoritmiske formuleringer

Afleveres på Blackboard

Frist aftales med instruktoren

Genaflevering (positivt) = instruktør feedback og mulighed for at forbedre sig
Anbefales grupper af 2-3 personer for løbende at kunne diskutere opgaver

Eksamen

2 timers multiple choice, uden hjælpemidler, 7-skala

Tidligere eksamensopgaver og træningsopgaver på kursussiden

Sprog

Forelæsninger og øvelser dansk
(materiale og kursusside dog engelsk)

Tidsforbrug	timer x uger
Forelæsninger	4 x 14 = 56
Teoretiske øvelser (TØ)	3 x 14 = 42
Studiecafé	1 x 14 = 14
Afleveringer	3 x 14 = 42
Forberedelse forelæsning	2 x 14 = 28
Forberedelse TØ	2 x 14 = 28
Forberedelse eksamen	45
Eksamen	2
Timer i alt	257

3 timer om ugen med en instruktør

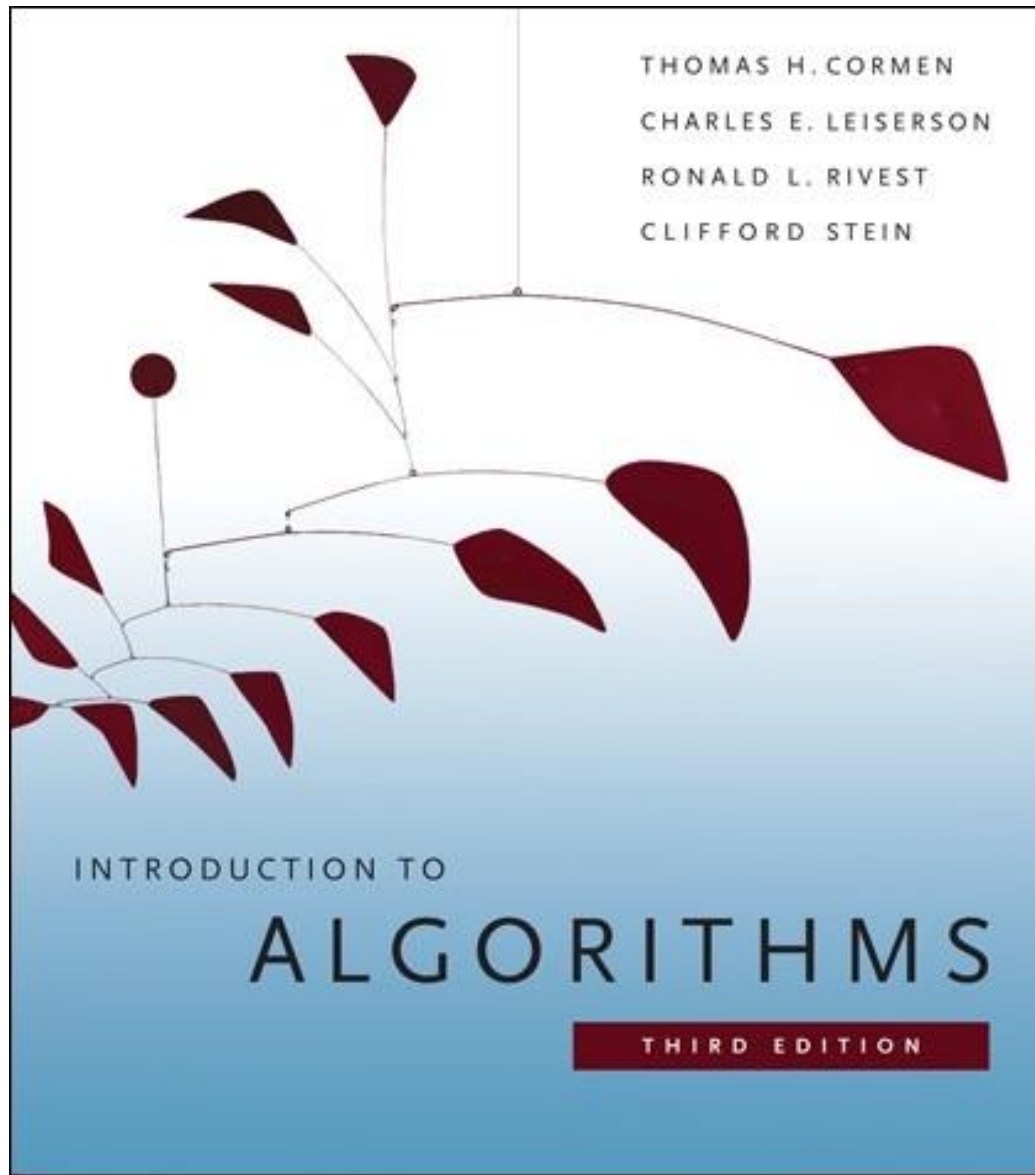
En række opgaver til hver gang

Opgaverne på “Course plan” hørende til forelæsninger *efter* jeres sidste TØ time, og *op til* denne TØ time

Format

1. I forbereder jer inden TØ i læsegrupperne og forsøger at forstå og løse alle opgaverne
Brug studiecafé & diskussionsforummet til hjælp
2. Ved øvelserne gennemgås opgaverne af de studerende, imens instruktoren hjælper med at rette misforståelser og fremhæve vigtige pointer

Sørg for løbende dialog med instruktoren om hvordan TØ timerne bruges bedst muligt



Primær lærebog

Supplementary Lecture notes

Algorithms and Data Structures

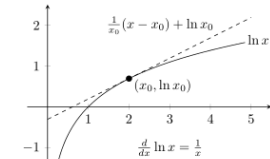
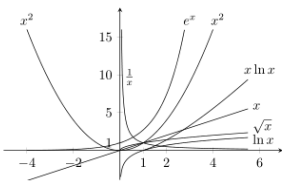
Gerth Stølting Brodal

Opdateres løbende igennem kurset

Et "cheat sheet" med matematisk notation og regneregler der anvendes i kurset findes i "Supplementary Lecture Notes"

Om 0 er med i de naturlige tal afhænger af hvem man spørger ! 

Mathematics Cheat Sheet

$\mathbb{N} = \{1, 2, 3, \dots\}$ natural numbers $\mathbb{R}$ = real numbers $\mathbb{R}^+ = \{x \in \mathbb{R} \mid x > 0\}$ <i>Floor</i> $\lfloor x \rfloor$, <i>ceiling</i> $\lceil x \rceil$ Associativity $(a \cdot b) \cdot c = a \cdot (b \cdot c)$, $(a + b) + c = a + (b + c)$ Commutativity $a \cdot b = b \cdot a$, $a + b = b + a$ Distributivity $a \cdot (b + c) = a \cdot b + a \cdot c$ Powers $a^0 = 1$, $a^1 = a$, $a^{b+c} = a^b \cdot a^c$ $(a^b)^c = a^{b \cdot c}$, $a^{b^c} = a^{(b^c)}$, $a^{-b} = \frac{1}{a^b}$ $(a \cdot b)^c = a^c \cdot b^c$, $a^{b-c} = a^b / a^c$ Roots $\sqrt[n]{a} = \sqrt[n]{a} = a^{1/n}$, $\sqrt[n]{a} \cdot \sqrt[n]{b} = \sqrt[n]{a \cdot b}$, $\sqrt[n]{a/b} = \sqrt[n]{a} / \sqrt[n]{b}$ Fractions $\frac{a}{b} \cdot \frac{c}{d} = \frac{a \cdot c}{b \cdot d}$ $\frac{a/b}{c/d} = \frac{a \cdot d}{b \cdot c}$ $\frac{a}{b} + \frac{c}{d} = \frac{a \cdot d + b \cdot c}{b \cdot d}$ Logarithms $\log_b a = c \Leftrightarrow b^c = a$, $b^{\log_b a} = a$ <i>Natural logarithm</i> $\ln a = \log_e a$, $e = 2.71828 \dots$ <i>Binary logarithm</i> $\log_2 a = \lg a$ $\log_b 1 = 0$, $\log_b b = 1$ $\log_b(a \cdot c) = \log_b a + \log_b c$ $\log_b(a/c) = \log_b a - \log_b c$ $\log_b(a^c) = c \cdot \log_b a$ $\log_b a = \frac{\log_c a}{\log_c b}$, $\log_b a = (\log_b a)^c$ $a^{\log_b c} = 2^{\log_2 a \cdot \log_2 c \cdot \frac{1}{\log_2 b}} = c^{\log_b a}$ 	Sets $Set A = \{a_1, a_2, \dots, a_n\}$ <i>Size</i> $ A $ <i>Membership</i> $x \in A$, <i>non-member</i> $x \notin A$ <i>Empty set</i> $\emptyset$, $ \emptyset = 0$ <i>Subset</i> $A \subseteq B$, i.e. $x \in A \Rightarrow x \in B$ <i>Set intersection</i> $A \cap B$ <i>Set union</i> $A \cup B$ <i>Set difference</i> $A \setminus B$ or $A - B$  $A \cup B = (A \setminus B) \cup (A \cap B) \cup (B \setminus A)$ <i>Commutativity</i> $A \cap B = B \cap A$, $A \cup B = B \cup A$ <i>Associativity</i> $A \cup (B \cup C) = (A \cup B) \cup C$ $A \cap (B \cap C) = (A \cap B) \cap C$ <i>Distributivity</i> $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$ $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$ <i>DeMorgan's laws</i> $A \setminus (B \cap C) = (A \setminus B) \cup (A \setminus C)$ $A \setminus (B \cup C) = (A \setminus B) \cap (A \setminus C)$ <i>Idempotence</i> $A \cup A = A$, $A \cap A = A$ <i>Empty set</i> $A \cup \emptyset = A$, $A \cap \emptyset = \emptyset$ <i>Complement</i> $\bar{A}$ wrt. <i>universe</i> U $\bar{\bar{A}} = U \setminus A$ where $A \subseteq U$, $\bar{\bar{A}} = A$ $\overline{A \cup B} = \bar{A} \cap \bar{B}$, $\overline{A \cap B} = \bar{A} \cup \bar{B}$ A and B are <i>disjoint</i> $\Leftrightarrow A \cap B = \emptyset$ <i>Cross product / Cartesian product</i> $A \times B = \{(a, b) \mid a \in A \wedge b \in B\}$ Sums ⁽²⁾ $\sum_{i=1}^n a_i = a_1 + a_2 + \dots + a_n$ $\sum_{i=1}^n (a_i + b_i) = \sum_{i=1}^n a_i + \sum_{i=1}^n b_i$ $\sum_{i=1}^n (c \cdot a_i) = c \cdot \sum_{i=1}^n a_i$ $\sum_{i=0}^n 2^i = 1 + 2^1 + \dots + 2^n = 2^{n+1} - 1$ $\sum_{i=0}^n a^i = \frac{a^{n+1} - 1}{a - 1}$ for $a \neq 1$ $\sum_{i=0}^{\infty} a^i = \lim_{n \rightarrow \infty} \sum_{i=0}^n a^i = \frac{1}{1-a}$, $ a < 1$ $\sum_{i=1}^n i = 1 + 2 + 3 + \dots + n = \frac{n(n+1)}{2}$ $\sum_{i=1}^n i^2 = \frac{n(n+1)(2n+1)}{6}$ $\sum_{i=1}^n i \cdot a^i = \frac{a}{(1-a)^2} (1 - a^n - n \cdot a^n + n \cdot a^{n+1})$ $\sum_{i=1}^{\infty} i \cdot a^i = \frac{a}{(1-a)^2}$ for $ a < 1$ <i>Polynomial</i> $P(x) = \sum_{i=0}^k c_i \cdot x^i$ <i>Telescoping sum</i> $\sum_{i=0}^{n-1} (a_{i+1} - a_i) = a_n - a_0$ n -te harmonic number $H_n = \sum_{i=1}^n \frac{1}{i} = \frac{1}{1} + \frac{1}{2} + \dots + \frac{1}{n}$ $\ln n + \frac{1}{n} \leq H_n \leq \ln n + 1$ $\lim_{n \rightarrow \infty} (H_n - \ln n) = \gamma = 0.577215 \dots$ = Euler-Mascheroni constant Products $\prod_{i=1}^n x_i = x_1 \cdot x_2 \cdot \dots \cdot x_n$ <i>Factorial</i> $n! = \prod_{i=1}^n i = 1 \cdot 2 \cdot \dots \cdot n$ $\ln(\prod_{i=1}^n x_i) = \sum_{i=1}^n \ln(x_i)$ $0 \leq \ln(n!) - (n \ln n - n + 1) \leq \ln n$	Probability theory ⁽³⁾ <i>Mean</i> $\bar{x} = \frac{x_1 + x_2 + \dots + x_n}{n}$ <i>Binomial coefficient</i> $\binom{n}{k} = \frac{n!}{(n-k)! \cdot k!}$ (number of ways to select k elements among n , independent on the order) <i>Linearity of expectation</i> $E[\sum_{i=1}^k c_i \cdot X_i] = \sum_{i=1}^k c_i \cdot E[X_i]$ <i>Bernoulli distribution</i> $X \sim \text{Bern}(p)$ $\Pr[X = 1] = p$, $\Pr[X = 0] = 1 - p$ <i>Binomial distribution</i> $X \sim \text{Bin}(n, p)$ (sum of n Bernoulli trials) $\Pr[X = k] = \binom{n}{k} \cdot p^k \cdot (1 - p)^{n-k}$ Expected value $\mu = E[X] = p \cdot n$ <i>Geometric distribution</i> $X \sim \text{Geom}(p)$ (number of Bernoulli trials before 1) $\Pr[X = k] = p \cdot (1 - p)^{k-1}$ $E[X] = \sum_{k=1}^{\infty} k \cdot \Pr[X = k] = \frac{1}{p}$ Logical expressions <i>Or / disjunction</i> $U \vee V$ <i>And / conjunction</i> $U \wedge V$ <i>Not / negation</i> $\neg U$ <i>Implication / conditional</i> $U \Rightarrow V$ <i>Equivalent / biconditional</i> $U \Leftrightarrow V$ <i>Exists</i> $\exists x : U(x)$ <i>For all</i> $\forall x : U(x)$ <table><tr><td>$\vee$</td><td>F</td><td>T</td><td>$\wedge$</td><td>F</td><td>T</td><td>$\neg$</td></tr><tr><td>F</td><td>F</td><td>T</td><td>F</td><td>F</td><td>F</td><td>F</td></tr><tr><td>T</td><td>T</td><td>T</td><td>T</td><td>F</td><td>F</td><td>T</td></tr></table> <table><tr><td>$\Rightarrow$</td><td>F</td><td>T</td><td>$\Leftrightarrow$</td><td>F</td><td>T</td><td>XOR</td><td>F</td><td>T</td></tr><tr><td>F</td><td>T</td><td>T</td><td>F</td><td>T</td><td>F</td><td>F</td><td>F</td><td>T</td></tr><tr><td>T</td><td>F</td><td>T</td><td>T</td><td>F</td><td>T</td><td>F</td><td>T</td><td>F</td></tr></table> F = false, T = true Asymptotic notation ⁽¹⁾ $f(x) = O(g(x)) \Leftrightarrow \exists c, x_0 \forall x \geq x_0 : f(x) \leq c \cdot g(x)$ $f(x) = \Omega(g(x)) \Leftrightarrow \exists c > 0, x_0 \forall x \geq x_0 : f(x) \geq c \cdot g(x)$ $f(x) = \Theta(g(x)) \Leftrightarrow f(x) = O(g(x)) \wedge f(x) = \Omega(g(x))$ Master Theorem ⁽¹⁾ Constants $a, c, d > 0, p \geq 0$ and $b > 1$ $T(n) = \begin{cases} c & \text{for } n \leq d \\ a \cdot T(n/b) + c \cdot n^p & \text{for } n > d \end{cases}$ $\Downarrow$ $T(n) = \begin{cases} \Theta(n^p) & \text{for } a < b^p \\ \Theta(n^p \cdot \log_b n) & \text{for } a = b^p \\ \Theta(n^{\log_b a}) & \text{for } a > b^p \end{cases}$ Growth of functions 	$\vee$	F	T	$\wedge$	F	T	$\neg$	F	F	T	F	F	F	F	T	T	T	T	F	F	T	$\Rightarrow$	F	T	$\Leftrightarrow$	F	T	XOR	F	T	F	T	T	F	T	F	F	F	T	T	F	T	T	F	T	F	T	F
$\vee$	F	T	$\wedge$	F	T	$\neg$																																												
F	F	T	F	F	F	F																																												
T	T	T	T	F	F	T																																												
$\Rightarrow$	F	T	$\Leftrightarrow$	F	T	XOR	F	T																																										
F	T	T	F	T	F	F	F	T																																										
T	F	T	T	F	T	F	T	F																																										
Literature	Mathematical formulas and terms (primary school), Ministry of Education, June 2017 [in Danish] Mathematical formulas (high school, stx A), Undervisningsministeriet, May 2018 [in Danish] Thomas H. Cormen <i>et al.</i> , Introduction to Algorithms, 3rd Edition, appendix A–C, 2009 Steve Seiden, Theoretical computer science cheat sheet, ACM SIGACT News, 27(4), 1996																																																	

Covered in ⁽¹⁾these notes, ⁽²⁾introduction to mathematics course, ⁽³⁾probability course, ⁽⁴⁾linear algebra course

Læringsmål fra kursusbeskrivelsen

I slutningen af kurset vil deltagerne kunne

- **Formulere** og **udføre** algoritmer og datastrukturer i form af pseudokode
- **Konstruere, implementere** og **analysere** algoritmer ved hjælp af standard algoritme paradigmer
- **Identificere** og **sammenligne** datastrukturer og grafalgoritmer til løsning af algoritmiske problemer
- **Identificere** gyldige invarianter for en algoritme
- **Konstruere, implementere** og **evaluere** algoritmers ydeevne for simple algoritmiske problemer
- **Analysere** og **sammenligning** tid og pladsforbrug af algoritmer og datastrukturer
- **Bevise** korrektheden af enkle programmer og transitionssystemer

Spørgsmål ?

Se Blackboard for info, samt slides!

Studieretning

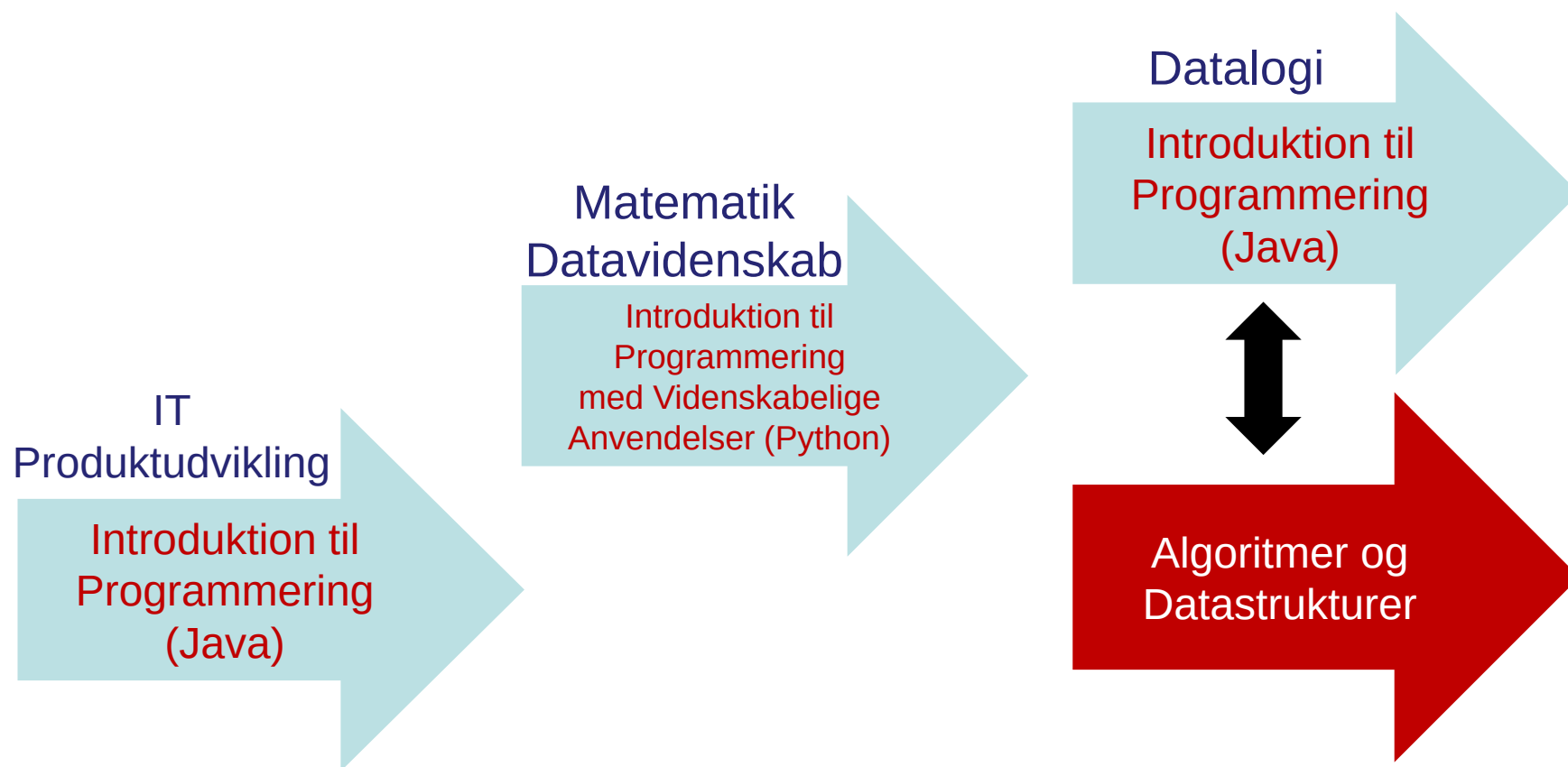
- a) Datalogi (1. år)
- b) Datavidenskab (2. år)
- c) IT produktudvikling (3. år)
- d) Matematik (typisk 3. år)
- e) Fysik (typisk 3. år)
- f) andet (typisk tilvalg i datalogi)

Programmeringserfaring

(for det programmeringssprog du måtte kende bedst)

- a) Ingen
- b) Basal kendskab
- c) Grundlæggende kendskab
- d) Avanceret
- e) Ekspert

- Algoritmer og Datastrukturer omhandler effektivitet af programmer
- Kræver egentlig man kan programmere...



- Programmeringsafleveringerne vil være basal Java

Teltstangs problemet

