

TEKNOLOGIFORSTÅELSE

8. KLASSE SOM FAG

3. forløb

TILFÆLDIGHEDER

- ALGORITMER OG DATA SOM BAGGRUND FOR FORUDSIGELSER

Udarbejdet af Malte von Sehested, Brian Ravnborg Christensen, Mikkel Hjort og Malene Erkmann *

*Materialet er udviklet af Københavns Professionshøjskole, Professionshøjskolen UCN, VIA University College samt læremiddel.dk for Børne- og Undervisningsministeriet under rammerne for Forsøg med teknologiforståelse i folkeskolens obligatoriske undervisning. Læs mere om forsøget på www.tekforsøget.dk og www.emu.dk.



KØBENHAVNS
PROFESSIONS
HØJSKOLE



LÆRE
MIDDEL
ØDK



VIA University
College

ucn

RAMBOLL

INDHOLDSFORTEGNELSE

1. Forløbsbeskrivelse	3
1.1 Overordnet beskrivelse – tre sammenhængende forløb.....	3
1.2 Resume.....	5
1.3 Rammer og praktiske forhold.....	5
2. Mål og faglige begreber.....	5
3. Forløbsnær del.....	7
3.1 Introfase: Forforståelse og kompetencer	7
3.2 Udfordrings- og konstruktionsfase.....	8
3.3 Outrofase: Ny forståelse og nye kompetencer.....	12
4. Perspektivering.....	13
4.1 Evaluering.....	13
4.2 Progression	13
4.3 Differentieringsmuligheder.....	13
4.4 Særlige opmærksomhedspunkter	13

Version 2

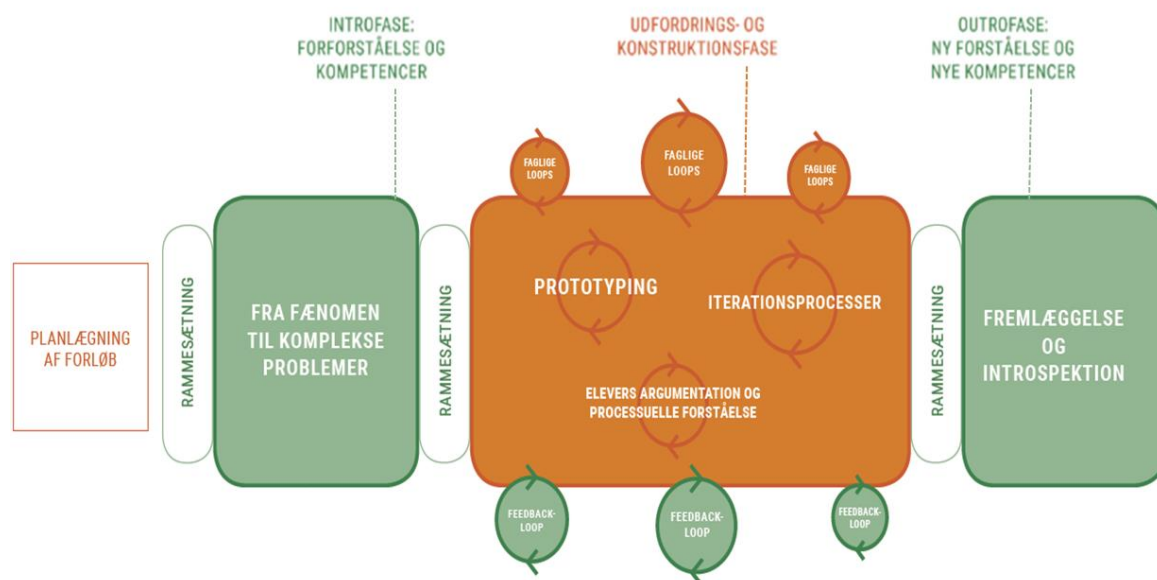
Dette er version 2 af forløbet. I revisionen af forløbene har vi arbejdet med at præcisere mål, rammer og aktiviteter. Der er ikke ændret fundamentalt ved forløbet, så materialer, som er udviklet til den konkrete undervisning på skolerne på baggrund af den første version af forløbet, vil stadig kunne anvendes.

1. Forløbsbeskrivelse

Dette forløb handler om de skjulte algoritmer, der styrer vores adfærd, og består af tre dele. Der tages udgangspunkt i simple spil som sten-saks-papir og gennem forløbet kommer eleverne til at arbejde med de data, der kan genereres i sådanne spil og med, hvordan dataene kan bruges til at modellere og forudsige fremtidige træk i spillet.

Forløbet har fokus på at give eleverne et grundlæggende kendskab til algoritmer og er bygget op med en introducerende del, en mere undersøgende/eksperimenterende del og en outro-del med opsamlings og evalueringer. Forløbet kommer derfor ikke igennem alle elementerne i det didaktiske format for prototyper, se figur 1, men kan ses som et fagligt loop, som giver mulighed for at arbejde mere problembaseret i det kommende forløb.

Figur 1: Forløbsmodel for prototyperne



1.1 Overordnet beskrivelse – tre sammenhængende forløb

Algoritmer og data benyttes i stigende grad af systemer der anvender forskellige grader af maskinlæring til, på baggrund af et stadig større datagrundlag, at forudsige sandsynligheder, muligheder og potentialer og i sidste ende til at foretage beslutninger. På et enkelt minut genereres uendelige mængder af data. Ifølge en opgørelse fra 2018 genereres fx 973.000 oplysninger om login på Facebook, der sendes 481.000 tweets og der bliver sendt 187 millioner e-mails, hvert eneste minut. Mange af disse data kan benyttes til at foretage forskellige forudsigelser og beslutninger.

En del af udfordringen i et digitaliseret samfund er at kunne forholde sig til hvordan smarte systemer og AI baseret på data og algoritmer foretager beregninger og beslutninger. Vi har brug for viden om, hvordan digitale systemer indsamler data om os, og hvordan vi selv kan indsamle og benytte data. Igennem tre forløb om algoritmer og forudsigelser arbejder eleverne gennem aktiviteter der skal udvikle deres faglige kompetencer og give dem færdigheder og viden til at forstå muligheder og konsekvenser i teknologier omkring os, der på baggrund af data og statistik træffer beslutninger om og for os. Eleverne skal gennem forløbene arbejde med teknologifagets fagbegreber, og der lægges i arbejdet med forløbene vægt på, at der bevidst arbejdes med at udvikle et nyt fælles sprog i klassen.

Forløbene er planlagt i følgende rækkefølge:

Tabel 2: Forløbsoversigt

TITEL OG VARIGHED	INDHOLD	KOMPETENCEMÅL
Tilfældigheder (8 lektioner)	I dette delforløb arbejder eleverne med at udvikle og spille sten-saks-papir. På baggrund af dataopsamling retter de herefter deres program, så det tager hensyn til fordelingen (sandsynligheden) for at der ikke er tale om et helt tilfældigt spil. Endelig perspektiveres i forhold til hvad det betyder, at man kender til sandsynligheden og hvordan det påvirker ens spil.	Eleven kan reflektere over og anvende computationel tankegang på problemstillinger fra omverdenen. Eleven kan vurdere, vælge og på kvalificeret vis anvende digitale teknologier i autentiske situationer
Fra tilfældighed til mønstre (18 lektioner)	I dette delforløb arbejder eleverne videre med spillet og indsamler data om hvordan sandsynligheden ændrer sig, afhængig af det foregående spil. Med udgangspunkt i et færdigt program arbejder de med at forstå hvordan spillet forandrer og vejer sine sandsynligheder. Endelig perspektiveres i forhold til at tage udgangspunkt i større datasæt og brug af hele spillet som baggrund for vejningen af sandsynligheder.	Eleven kan reflektere over og anvende computationel tankegang på problemstillinger fra omverdenen. Eleven kan vurdere, vælge og på kvalificeret vis anvende digitale teknologier i autentiske situationer Eleven kan handle med dømmekraft i komplekse situationer, der vedrører digitale artefaktens betydning for individ, fællesskab og samfund
Udfordring (14 lektioner)	I tredje del skal eleverne bruge deres erfaringer med at indsamle data til at omforme et nyt spil til data, der kan danne baggrund for forudsigelser, som eleverne kan bruge til at vinde i spillet.	Eleven kan reflektere over og anvende computationel tankegang på problemstillinger fra omverdenen. Eleven kan vurdere, vælge og på kvalificeret vis anvende digitale teknologier i autentiske situationer

TITEL OG VARIGHED	INDHOLD	KOMPETENCEMÅL
		Eleven kan handle med dømmekraft i komplekse situationer, der vedrører digitale artefaktens betydning for individ, fællesskab og samfund

1.2 Resume

I dette delforløb arbejder eleverne med det allerede velkendte spil: sten-saks-papir. De starter med at udvikle et program der kan bruges til at spille spillet ud fra principper om tilfældighed. På baggrund af dataopsamling retter de herefter deres program, så det tager hensyn til fordelingen (sandsynligheden) for, at der ikke er tale om et helt tilfældigt spil. Endelig perspektiveres i forhold til, hvad det betyder, at man kender til sandsynligheden, og hvordan det påvirker ens spil.

Produkt

Udgangspunktet for forløbet er en programmeret udgave af sten-saks-papir, som eleverne udvikler i Python eller Scratch. Programmet vil i første omgang være baseret på tilfældighed, men vil blive ændret på baggrund af dataindsamlingen.

1.3 Rammer og praktiske forhold

Forløbet kræver ingen særlige lokaler eller andet udstyr end almindelige computere.

1.3.1 Samlet varighed

8 lektioner svarende til ca. 4 uger - afhængigt af brugen af faglige loops.

- Introfase (1 lektion)
- Udvikling af sten-saks-papir spil med tilfældigheder (2 lektioner)
- Spil, dataopsamling og databehandling (2 lektioner)
- Udvikling af sten-saks-papir spil med sandsynligheder (2 lektioner)
- Perspektivering (1 lektion)

1.3.2 Materialer

Der vil under forløbet være brug for følgende ressourcer:

- Computer med adgang til internettet
- Regneark til dataopsamling

2. Mål og faglige begreber

I denne del skal eleverne opnå en begyndende forståelse for, hvordan algoritmer kan bruges til at forudsige andres handlinger og til at vinde i spil. Eleverne skal gennem dataopsamling opnå en forståelse for, hvordan et spil kan omsættes til data for en computer, og de skal træne programmering af algoritmer.

Endelig kan eleverne bruge deres algoritmer som udgangspunkt for at diskutere, hvordan sådanne algoritmer påvirker verden omkring dem - f.eks. ift. online gambling, aktiekurser osv.

KOMPETENCEOMRÅDER	COMPUTATIONEL TANKEGANG	TEKNOLOGISK HANDLEEVNE
Kompetencemål (efter 9. klassetrin)	Eleven kan reflektere over og anvende computationel tankegang på problemstillinger fra omverdenen	Eleven kan vurdere, vælge og på kvalificeret vis anvende digitale teknologier i autentiske situationer
Færdigheds- og vidensmål (efter 9. klassetrin)	<p>Data</p> <p>Eleven kan behandle, vurdere og visualisere data reflekteret ved hjælp af digital teknologi.</p> <p>Eleven har viden om kriterier for datakvalitet</p> <p>Algoritmer</p> <p>Eleven kan vurdere forskellige algoritmers anvendelighed og kan benytte forskellige metoder til at afprøve algoritmer.</p> <p>Eleven har viden om forskellige parametre til vurdering af algoritmers anvendelighed</p> <p>Modellering</p> <p>Eleven kan konstruere digitale modeller af virkeligheden og ud fra dem lave forudsigelser og følgeslutninger og vurdere begrænsninger i modellen.</p> <p>Eleven har viden om, hvordan abstraktion af virkeligheden kan bruges til at beskrive og behandle denne i digitale modeller, og hvordan man kan afprøve en model ift. dens intentioner.</p>	<p>Programmering</p> <p>Eleven kan læse og forstå programmer skrevet i et tekstbaseret programmeringssprog samt anvende et sådant til systematisk modifikation og konstruktion af programmer ud fra en problem-specifikation.</p> <p>Eleven har viden om metoder til at analysere og forudsige programmers opførsel samt teknikker til systematisk og trinvis udvikling af programmer.</p>

Konkretiserede læringsmål. Eleven kan...

- udvælge og skabe relevante data for sten, saks, papir
- programmere algoritmen
- vurdere algoritmens anvendelighed og forbedringsmuligheder

3. Forløbsnær del

3.1 Introfase: Forforståelse og kompetencer

Eleverne arbejder i denne del med udvikling af deres eget program baseret på tilfældighed. Der arbejdes således med teknologisk handleevne og algoritmisk forståelse.

Der arbejdes med begreber som sandsynlighed, tilfældighed, betingelser og dataopsamling. Det anbefales, at arbejdet begynder med en fælles snak i klassen om de centrale begreber og med at opbygge algoritmer til at brug for at spille sten-saks-papir baseret på tilfældighed.

3.1.1 Varighed

Estimeret til 1 lektion a 45 minutter.

3.1.2 Problemfelt

Sådan som vores samfund er indrettet i dag, styres en stor del af processerne af algoritmer. Det er ikke som sådan noget nyt, fordi der altid har været procedurer og systemer til at håndtere menneskelig interaktion, men algoritmerne har enorme potentialer til at forandre vores samfund. I dette forløb undersøger eleverne algoritmer i spillet sten, saks, papir som et eksempel på, hvordan algoritmer kan bestemme udfald af spil og dermed af menneskelig interaktion.

3.1.3 Problemstilling

Eleverne arbejder i dette forløb med at forstå, hvordan algoritmer kan påvirke menneskelige interaktioner gennem at undersøge problemstillingen: Hvordan kan vi fremstille et program der i stand til at spille sten-saks-papir ud fra principper om tilfældighed og på baggrund af en databaseret undersøgelse udvide og ændre vores program?

3.1.4 Iscenesættelse/scenarie:

Computere er gode til at følge regler og rutiner. Ved at lave programmer kan man få computeren til at følge en række instruktioner. Så når man kan nedbryde en opgaveløsning til en algoritme, kan man også skrive et program der kan få en computer til at følge denne algoritme. De fleste af de programmer, vi benytter i dagligdagen, fungerer på denne måde, hvad enten det er til at skrive og sende e-mail, lave et opslag på Facebook eller se balancen på bankkontoen: Der er regler for, hvordan en e-mail sendes og modtages, på Facebook er der et system til, hvordan input fra brugeren bliver til et opslag og udregningen af bankkontoens balance baserer sig på (relativt) simple beregninger.

Problemet er, at vi ikke altid ved, hvordan noget præcist skal gøres, og når vi ikke kender reglerne, er det meget sværere at udføre en opgave. Hvis vi ikke er helt sikre på, hvordan en række af regler fungerer, er det jo svært at fortælle en computer, hvordan den skal gøre det. Det kan også være at opgaveløsningen er

baseret på usikkerhed. Når vi møder usikkerhed, prøver vi at lave det bedste gæt. Et gæt baseret på vores erfaringer og tanker om sandsynligheder for de forskellige udfald.

I forhold til et simpelt spil som sten-saks-papir kan vi naturligvis bruge den viden at fordelingen af træk ikke er lige. Der er flere der spiller sten end henholdsvis saks og papir. Umiddelbart betyder det, at den bedste strategi til at vinde er altid at spille papir. Problemet er bare, at hvis vores modstander har gennemskuet dette, er vi lette at slå. Derfor bliver vores bedste træk at basere det på det vi allerede ved om modstanderen, og på det eller de træk, der lige er lavet.

Begynd arbejdet med at tale om de centrale begreber i forløbet: sandsynlighed, tilfældighed, betingelser og dataopsamling. Hvilke elever kender begreberne, og hvad forstår de ved dem. Eleverne kan først diskutere det to og to, og derefter dele deres snak med klassen. Snak herefter med eleverne om simple spil. Hvornår er det tilfældigheder, som afgør, hvem der vinder, og hvornår kan man have fordel af at regne sandsynligheder.

Man kan herefter starte med at lade eleverne spille sten-saks-papir mod hinanden eller eventuelt besøg <https://archive.nytimes.com/www.nytimes.com/interactive/science/rock-paper-scissors.html> og se, hvem der klarer sig bedst.

3.2 Udfordrings- og konstruktionsfase

I denne del arbejder eleverne med at programmere samt med at indsamle og behandle data. Det anbefales at eleverne arbejder i grupper.

3.2.1 Varighed

I alt estimeret til 4 lektioner a 45 minutter. Dertil kommer faglige loops, hvis behov afhænger meget af elevernes erfaringer.

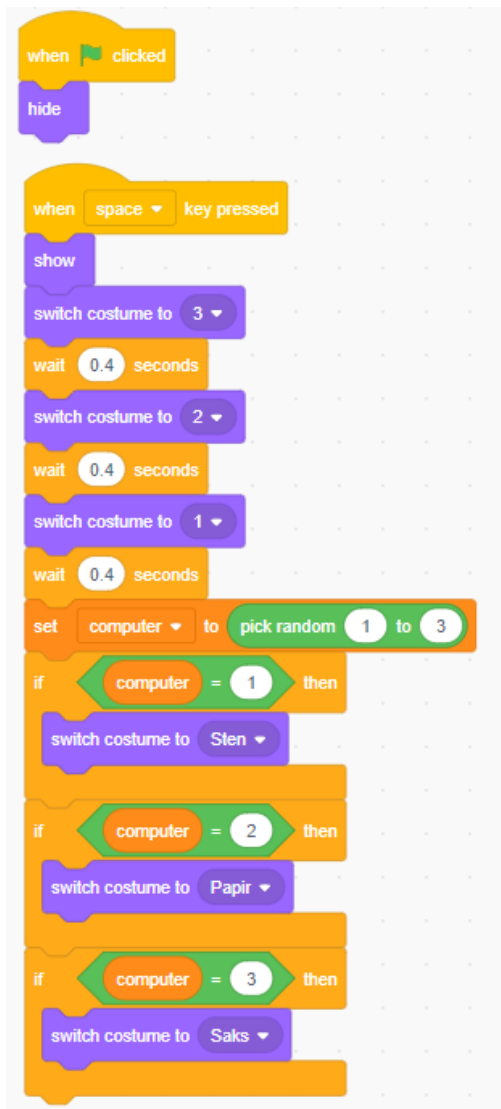
3.2.2 Konkrete udfordringer

Udvikling (2 lektioner)

Der arbejdes i enten Scratch eller Python med at skrive et simpelt program der kan håndtere, at man kan spille sten-saks-papir. Programmet skal give spilleren mulighed for at vælge og skal (uden at snyde) komme med programmets bud, samt afgøre hvem der har vundet.

Eksempel på program i Scratch

Programmet kræver, at man har "kostumer" med tallene fra 1-3 samt "figurerne" sten, saks og papir. Spillet afgør ikke, hvem der har vundet, men lader blot computeren spille. En anden måde at lave programmet på er, at lade spilleren vælge sit træk og så lade computeren vælge sit og vise, hvem der har vundet.



Eksempel på program i Python

```
#!/bin/python3

from random import randint

# Spillerens træk

spiller = input('Vælg sten (1), saks (2) eller papir (3)')
spiller = int(spiller)

if (spiller == 1):
    print('Sten',end=' ')
elif (spiller == 2):
    print('Saks',end=' ')

```

```

elif (spiller == 3):
    print('Papir',end=' ')

print('mod',end=' ')

# Computerens træk

computer = randint(1,3)

if (computer == 1):
    print('Sten')
elif (computer == 2):
    print('Saks')
elif (computer == 3):
    print('Papir')

## Afgøre hvem der vandt

if (spiller == computer):
    print('Uafgjort')
elif (spiller == 1 and computer == 2):
    print('Spiller vinder')
elif (spiller == 1 and computer == 3):
    print('Computer vinder')
elif (spiller == 2 and computer == 3):
    print('Spiller vinder')
elif (spiller == 2 and computer == 1):
    print('Computer vinder')
elif (spiller == 3 and computer == 1):
    print('Spiller vinder')
elif (spiller == 3 and computer == 2):
    print('Computer vinder')

```

3.2.2.1 Dataopsamling (1 lektion)

Der arbejdes nu med at spille spillet og opsamle data. De vigtige data er, hvem der har spillet hvad. Til opsamling kan bruges en simpel tabel:

SPILNUMMER	SPILLER	COMPUTER
1	Sten	Saks
2	Sten	Papir
3	Saks	Sten
...

3.2.2.2 Databehandling (1 lektion)

Databehandlingen handler om se på hyppigheder. Er der nogle træk der bliver brugt oftere end andre og hvad betyder det for den måde som man bør spille på? Ved at lave frekvensanalyse på de indsamlede data, skal eleverne nu lave en strategi for, hvordan de kan gøre deres program bedre i stand til at vinde.

Hvis det for eksempel viser sig at spillerne oftere spiller saks end andre træk, kan man så bruge dette til noget?

3.2.2.3 Bedre program (2 lektioner)

Eleverne skal nu forbedre deres program baseret på deres data. Så i stedet for at det er tilfældigt, hvilket træk computeren laver, skal den lave et træk på baggrund af de indsamlede data.

Eleverne må selv udvikle en strategi. Det kan for eksempel være at deres data viser, at der spilles sten 50% af tiden, hvorfor de laver en ændring så deres program altid spiller papir, eller de måske ændrer programmet så det oftere spiller papir, men stadig er baseret på en vis grad af tilfældighed, ved at øge sandsynligheden for netop papir.

3.2.3 Faglige loops

Afhængigt af elevernes programmeringsmæssige erfaringer, kan man have behov for at arbejde med sandsynlighed, variable og betingelser.

3.2.4 Feedbackloops

Der kan arbejdes med et kort feedbackloop i forbindelse med forløbet

Feedback efter databehandlingen

Når grupperne er færdige med databehandlingen, kan de diskutere deres resultater med en eller flere andre grupper, eller diskussionen kan tages i klassen. Her kan tages udgangspunkt i følgende spørgsmål, først til selve resultaterne og derefter til arbejdets relevans i samfundet:

- Hvilke tendenser så vi, da vi analyserede vores data?
- Var det overraskende?
- Hvad gav det anledning til at justere?
- Vil det nye program vinde oftere?

- Hvilke eksempler kan vi komme på, hvor algoritmer styrer ting, som foregår i vores hverdag?
- Kan I komme i tanker om algoritmer, som løbende forbedres på baggrund af de data, som man som bruger bidrager med (fx søgninger på Google, videoer på Youtube, nyheder på Facebook)?
- Er det altid en forbedring, når en algoritme tager hensyn til ens historik (I forløbets eksempel det man tidligere har spillet, men det kunne også være, det man tidligere har søgt på)?

3.3 Outrofase: Ny forståelse og nye kompetencer

3.3.1 Varighed

Estimeret 1 lektion á 45 minutter.

3.3.2 Fremlæggelse og introspektion

Når grupperne er færdige med deres første program, kan de præsentere det for en anden gruppe og få feedback til deres løsning. En måde at arbejde med denne feedback er med fokus på systemundersøgelse og systemforståelse:

Responsgruppen starter med at undersøge den anden gruppes program. Efter at have undersøgt programmet gives en kortvarig mulighed for at spørge til, hvis der er dele af programmet, som ikke umiddelbart er forstået af responsgruppen. Herefter bruger responsgruppen tid på at forberede en feedback i forhold til det sted i programmet, hvor de ser det største potentiale for forbedring.

4. Perspektivering

4.1 Evaluering

Hovedfokus i forløbet er elevernes viden og færdigheder ift. data og algoritmer. Læreren vurderer undervejs ifm. både skabelse og test af algoritmerne, i hvor høj grad eleverne er i stand til at udvælge og skabe relevante data for deres spil. I forbindelse med outro-fasen vurderer læreren i særlig grad, i hvor høj grad eleverne er i stand til at forstå og forholde sig til andres algoritmer - deres anvendelighed og forbedringsmuligheder.

4.2 Progression

Eleverne bygger i dette forløb videre på det arbejde, de har lavet med programmering, data og algoritmer siden 1. klasse. Forløbet bygger videre på og forudsætter mange af de kompetencer og færdigheder, eleverne har erhvervet sig i hele arbejdet med teknologiforståelse.

4.3 Differentieringsmuligheder

Elevernes arbejde kan differentieres ved at styre deres valg af enten blok- eller tekst-baseret programmering. Denne differentiering vil kunne følge grupperne til Del 2 og 3.

4.4 Særlige opmærksomhedspunkter

Dette forløb trækker på mange af de færdigheder og kompetencer, eleverne har opøvet igennem deres arbejde med teknologiforståelse, og derfor vil det for nogle elever være et vanskeligt forløb. Forløbet stiller store krav til lærerens evne til selv at se mulighederne for at anvende algoritmer til at vinde spil. Det er væsentligt at læreren lader eleverne prøve sig frem - også selv om resultatet ikke bliver lige godt hver gang.