

TEKNOLOGIFORSTÅELSE MELLEMTIN

5. ÅRGANG

1. forløb: Eleverne som kunstnere



KØBENHAVNS
PROFESSIONS
HØJSKOLE



LÆRE
MIDDEL
ØDK



VIA University
College



INDHOLDSFORTEGNELSE

1. Forløbsbeskrivelse	3
1.1 Beskrivelse	3
1.2 Rammer og praktiske forhold.....	5
2. Mål og faglige begreber.....	6
3. Forløbsnær del.....	8
3.1 Introfase: Forforståelse og kompetencer	8
3.2 Udfordrings- og konstruktionsfase.....	9
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
5. Udvidelse af forløbet.....	14

1. Forløbsbeskrivelse

1.1 Beskrivelse

1.1.1 Fælles introduktion til tre forløb i 5. klasse: Kunst, apps og hjemmesider

En kompleks problemstilling som overordnet tema:

Efterårets forløb til 5. klasse er bygget op af tre mindre forløb, som alle overordnet arbejder med problemstillingen: Mennesker, maskiner: magt og samarbejde.

Teknologien har været menneskets hjælpemiddel og slave i århundreder, men nærmer vi os en tid, hvor maskinen bliver vores ligemænd, og hvor meget styrer vi egentlig selv teknologien?

I dette undervisningsforløb arbejder eleverne med komplekse spørgsmål om mulighederne for menneskets samarbejde med maskinen – eller teknologierne – både i forhold til:

- kunst og menneskers brug af teknologi i kunstnerisk produktion og processer
- udarbejdelse af skitser til design for apps til gavn for andres konkret formulerede problemstillinger i hverdagen, (hvordan skabes design til gavn for mennesker?)
- i forståelser, afkodning og design af hjemmesider på nettet.

De tre didaktiske forløb rammesætter dermed tre måder at anskue det komplekse problem på; om det er mennesker eller maskiner der har magten? De didaktiske forløb giver bud på nogle tilgange til problemstillingen, men de tæller ikke som entydige svar. Målet er, at forløbene hver især eller som ét samlet forløb skal give eleverne mulighed for opnå viden til foreløbige løsninger, så de kan give nogle bud på rimelige vurderinger i arbejdet med det komplekse problem. På den måde kan eleverne i fællesskab arbejde sig hen mod en større og mere stabil forståelse af, at teknologier og maskiner i dag fylder meget i vores hverdag, og måske også til tider fylder for meget ift. hvem der har kontrollen?

Synliggørelse af kompleks problemstilling i de tre didaktiske forløb:

I det første forløb eksperimenterer eleverne med kunst og algoritmer og har primært fokus på computationel tankegang og konstruktion af digitale artefakter ved hjælp af programmering. Inden for de æstetiske domæner arbejdes der som i alle andre domæner med de nye muligheder, som det computationelle giver, bl.a anvendes kunstig intelligens i forskellige musik-, billed- og litteraturskabende processer, og der findes et relativt nyt fagfelt på forskellige universiteter i verden, som kaldes "computationel kreativitet", som netop arbejder med koblinger mellem det æstetiske og det computationelle.

Det næste forløb fokuserer på designprocesser i forbindelse med udformning af apps.

Her afprøver eleverne og arbejder sig igennem en model for designforløb. Eleverne lærer at udvælge et komplekst problemfelt, som kan viderebearbejdes og formuleres som et problem. Dette mere konkrete problem kan der udarbejdes løsninger på gennem inddragelse af teknologi, så denne bidrager til, men i lige så høj grad også indflyder på løsningen. Gennem feedback-loops fremlægger eleverne for hinanden, afprøver og giver feedback samt argumenterer og reflekterer over deres valg og designprocesser.

Det sidste forløb fokuserer på hjemmesiders struktur og virkning. Vi arbejder med hjemmesider ud fra to perspektiver, både "udefra" og "indefra", og i den sammenhæng fokuseres der på datastrukturer og

modellering af data. Disse processer kommer særligt til syne, når eleverne både skal manipulere med hjemmesider samt lave deres egen hjemmeside i en opsat skabelon. Arbejdsformen er kendetegnet af, at eleverne skal arbejde med at remixe "koder", som bestemmer hvordan resten af hjemmesiden virker. Dette afføder spørgsmål om, hvad vi som modtagere af hjemmesider skal være bevidste om - bliver vi nogle gange snydt, eller er afsenderen troværdig?

Selvom de tre didaktiske forløb alle kan perspektiveres til ovenstående fælles problemstilling, kan de også fint fungere som tre individuelle forløb, da de har hvert deres fokus og bruger forskellige teknologier.

1.1.2 Introduktion til forløb 1: Eleverne som kunstnere

Dette forløb handler om primært om computationel tankegang og konstruktion af digitale artefakter ved hjælp af programmering. Det digitale artefakt, som eleverne skal konstruere, er et interaktivt kunstværk. Forløbet er indsat i en scenariedidaktisk ramme, som lægger op til, at eleverne arbejder i mindre grupper med deres egne udtryk undervejs i opgaverne. Eleverne arbejder med at omsætte idéer til skitser og algoritmer, og derfra til en programmeret løsning, som i høj grad bygger på dekomposition og mønstergenkendelse.

I forløbet anvendes Scratch eller CodingLab, som er Scratch til skoletube. Undervejs anvender eleverne kontrolstrukturer, som er generelle for kodning, uanset programmeringssprog og – platform.

I slutningen af forløbet skal eleverne reflektere over, om computerkunst er kunst og derigennem forholde sig til forholdet mellem menneske og maskine. På den ene side strækker det computationelle den menneskelige formåen, og på den anden side kan der findes eksempler på computerkunst, som er så autogenereret, at det (måske?) holder op med at være kunst, og blot med få linjer kode bliver rå visualiseringer af data. Hvori ligger så det menneskelige og det kreative?

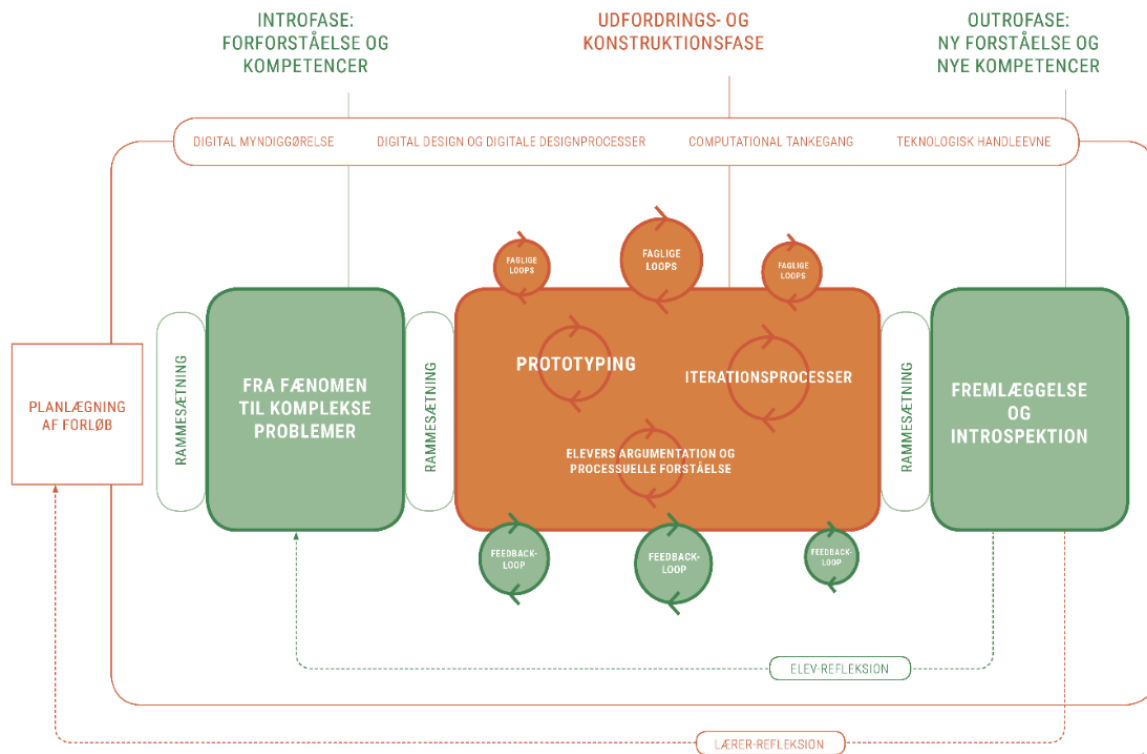
På et mere nært niveau skal eleverne forholde sig til kunstnerens autonomi og ret til at bestemme, hvorledes et værk skal udtrykkes og fortolkes, i og med at en digitalisering af et værk gør det muligt at omdefinere og grundlæggende ændre dette.

1.1.3 Produkt

Med udgangspunkt i en algoritme, som er lavet af en virkelig kunstner og udført på forskellige museer i verden, skal eleverne producere deres egne versioner af dette værk.

Forløbet er bygget op over det didaktiske format for prototyperne med en introducerende del, en mere undersøgende/eksperimenterende del og en outro-del med opsamlinger og evalueringer, se figur 1.

Figur 1: Forløbsmodel for prototyperne



1.2 Rammer og praktiske forhold

1.2.1 Samlet varighed

Hele forløbet er estimeret til en varighed på ca. 15 lektioner á 45 minutters varighed. Der er dog indlagt valgfrie moduler, hvis læreren ønsker at forkorte forløbet for at give plads til andre aktiviteter, som ønskes lokalt. Omvendt vil de enkelte moduler også kunne udvides, så forløbet varer længere, fx ved at lade eleverne lave deres helt eget kunstværk og tilhørende algoritme til sidst.

1.2.2 Materialer

Eleverne skal have adgang til computere.
Derudover smartboards, projektor eller lign. til præsentationer

- Forslag til præsentation – kan remedieres efter behov. Her er undervejs indlagt små videoer, der rammesætter det scenariedidaktiske forløb.
- Introduktionsøvelser til penneværktøj
- Kopi af "Sol LeWitts brev" med algoritmen for Wall Drawing #273
- Elevark til skitsetegninger

- Elevark: Worked example til udformning af første væg
- Elevark: Forslag til udformning af efterfølgende vægge
- Tusser eller farveblyanter (rød, blå, gul) samt evt. linealer
- Færdige programmer, der kan modificeres, og som læreren kan vælge at dele med elever, hvis der er behov.

1.2.3 Lokaler

Der kan arbejdes i klasselokalet, men gruppe lokaler eller -faciliteter vil gavne aktiviteterne i høj grad.

1.2.4 Videnspersoner og andre eksterne aktører

Det er en mulighed, men ikke nødvendigt, at udvide forløbet med interviews eller besøg hos videnspersoner eller kulturinstitutioner i nærområdet, som beskæftiger sig med feltet.

1.2.5 Tværfaglighed

Der er oplagte muligheder for at arbejde tværfagligt med henholdsvis billedkunst og matematik

2. Mål og faglige begreber

2.1.1 Computational tankegang:

I forløbet fokuseres på elevernes evne til at læse, omsætte og videreudvikle algoritmer i deres arbejde med at digitalisere kunstværket. For at skabe deres løsninger, skal eleverne nedbryde processen i håndtérbare delelementer og anvende den samme løsning med tilretninger for at udføre det samlede værk, altså arbejde med mønstergenkendelse. Herigennem opfyldes målparrene under **"Algoritmer"** og **"Strukturering"**. Forløbet indeholder desuden både håndtegnede skitser og digitale modeller af det samme, og eleverne skal reflektere over muligheder og begrænsninger. Dette peger ind i målparret under **"Modellering"**, om end dette kun berøres i en simpel form. Alt i alt arbejdes der dog i høj grad med kompetencemålet for "Computational tankegang".

2.1.2 Teknologisk handleevne:

Gennem arbejdet trænes elevernes programmeringsfærdigheder i blokprogrammering samt afprøvning og fejlretning. Der arbejdes kun med færdigheds-/vidensområdet **"Programmering"** under dette kompetenceområde, og dermed primært med den del af kompetencemålet, som omhandler viden om digitale teknologiers sprog og principper.

2.1.3 Digital design og designprocesser:

Eleverne arbejder med at udtrykke deres idéer digitalt og reflekterer over, hvad det egentlig betyder, at man kan digitalisere kunst for kunstneren og værket i sig selv, altså digitale teknologiers anvendelsesmuligheder og artefakters anvendelse (jf. den komplekse problemstilling om forholdet mellem mennesker og maskiner). Dermed peger forløbet ind i målparrene under **"Konstruktion"**. Forløbet afsluttes med fernisering, hvor eleverne argumenterer for deres løsninger og redegør for deres designprocesser og beskæftiger sig dermed også med målparrene under **"Argumentation og introspektion"**. Der er dog ikke primært fokus på selve designprocessen i forløbet, det følger i næste forløb omkring Apps. Det komplekse problemfelt om mennesker og maskiner danner en overordnet perspektiveringsramme,

men anvendes ikke direkte i elevernes egne løsninger, så samlet arbejdes der kun i lille grad med kompetencemålet for "Digital design og designprocesser".

KOMPETENCE-OMRÅDER	COMPUTATIONEL TANKEGANG	TEKNOLOGISK HANDLEEVNE	DIGITAL DESIGN OG DESIGNPROCESSE
Kompetencemål (efter 6. klassesetrin)	Eleven kan følge og anvende computationel tankegang i arbejdet med konkrete problemstillinger	Eleven kan, med udgangspunkt i viden om digitale teknologiers sprog og principper, handle med overblik med digitale teknologier i konkrete situationer	Eleven kan skabe digitale artefakter med digitale teknologier og gennemføre iterative designprocesser, der løser komplekse problemstillinger, relevante for individ og fællesskab
Færdigheds- og vidensmål (efter 6. klassesetrin)	<p>Algoritmer</p> <ul style="list-style-type: none"> Eleven kan genkende og tilrette algoritmer i forskellige sammenhænge og redegøre for deres funktion Eleven har viden om kendetegn ved algoritmer og deres opbygning, samt hvordan de anvendes i forskellige sammenhænge 	<p>Programmering</p> <ul style="list-style-type: none"> Eleven kan beskrive, tilrette og konstruere programmer i blokbaserede programmeringssprog samt foretage systematisk afprøvning og fejlretning af egne og andres programmer Eleven har viden om konstruktioner i blokbaserede programmeringssprog og teknikker til systematisk konstruktion, fejlfinding og fejlretning af programmer 	<p>Konstruktion</p> <ul style="list-style-type: none"> Eleven kan med digitale teknologier konstruere artefakter, som udtrykker en idé, Eleven har viden om konstruktion af artefakter og om digitale teknologiers anvendelsesmuligheder og kan reflektere over artefaktets anvendelse
	<p>Strukturering</p> <ul style="list-style-type: none"> Eleven kan anvende mønstre i strukturering af data og dataproceser med udgangspunkt i konkrete problemstillinger Eleven har viden om mønstre i strukturering af data og dataproceser 		<p>Argumentation og introspektion</p> <ul style="list-style-type: none"> Eleven kan argumentere for sammenhænge mellem rammesættelse, idégenerering og konstruktion og kan forholde sig til egen designkompetence Eleven har viden om fagtermer for

KOMPETENCE-OMRÅDER	COMPUTATIONEL TANKEGANG	TEKNOLOGISK HANDLEEVNE	DIGITAL DESIGN OG DESIGNPROCESSER
			argumentation om designprocesser og for egen designkompetence
	Modellering <ul style="list-style-type: none"> ■ Eleven kan anvende digitale modeller i forskellige faglige sammenhænge og i arbejdet med konkrete problemstillinger ■ Eleven har viden om, hvordan forskellige modeller kan beskrive samme virkelighed, samt muligheder og begrænsninger ved forskellige modeller 		

3. Forløbsnær del

I forløbet skal eleverne forestille sig, at de er ansatte på et museum, som har fundet et gammelt brev fra en kunstner. Kunstneren, der tages udgangspunkt i, Sol LeWitt, er en anerkendt kunstner, som formulerede sine værker som algoritmer, der kunne udføres af museer over hele verden. I det konkrete forløb anvendes Wall Drawing #273, som er en forholdsvis simpel algoritme, som samtidig har mange "tomme" felter, eleverne selv skal udfylde.

Eleverne guides gennem forløbet af en "museumsdirektør" og møder også til sidst en "kunstkritiker" gennem små videoer. Klassens lærer(e) kan naturligvis vælge selv at spille rollerne, hvis dette foretrækkes. Efter hovedforløbet er der mulighed for at udvide med et designforløb, hvor eleverne konstruerer deres helt egne kunstprojekter. Der er skitseret et par muligheder for dette under punkt 5.

3.1 Introfase: Forforståelse og kompetencer

For at eleverne kan gennemføre forløbet, skal de kende penneværktøjet i Scratch. I introfasen arbejdes der derfor med små opgaver som har til formål at sikre, at alle elever kan anvende dette. Kan eleverne ikke bruge værktøjet i Scratch, vil det være besværligt at gennemføre forløbet. I den efterfølgende udfordrings- og konstruktionsfase arbejder eleverne scenariedidaktisk med at konstruere deres digitale artefakter og først til sidst sættes forløbet ind i en større ramme. Dette er et bevidst valg, da eleverne dermed kan anvende deres egne erfaringer i overordnede og abstrakte refleksioner vedrørende problemfelt og problemstillinger.

3.1.1 Varighed

Estimeret til 2 lektioner á 45 minutter.

3.1.2 Komplekst problemfelt

Der arbejdes overordnet i hele forløbet med det komplekse problemfelt "mennesker, maskiner, magt og samarbejde". Det komplekse problemfelt spiller dog ikke en central rolle i introfasen (som prototypemodellen ellers peger på, se pkt. 1.1.3.), da det vurderes, at det er vigtigere at eleverne gøres fortrolige med penneværktøjet i Scratch, således at de kan gennemføre den efterfølgende udfordrings- og konstruktionsfase og outfase. Undervejs får eleverne viden om, hvorledes der arbejdes og samarbejdes med "maskiner" i forhold til æstetiske domæner, og sidst i forløbet vil eleverne skulle forholde sig til, om computerskabt kunst egentlig er kunst. Dermed fylder det komplekse problemfelt betydeligt mere i outfase end introfasen i dette forløb.

På et mere lavpraktisk plan kan der argumenteres for, at Sol LeWitts algoritme i sig selv er et komplekst problem, i og med at der er mange "tomme felter" som eleverne selv skal udfylde. Der er altså mange mulige løsninger, der vil kunne anvendes.

3.1.3 Problemstilling

Hvordan er computerskabt kunst, kunst? Kan algoritmer lave kunst? Er der forskel på kunst lavet af maskiner frem for mennesker?

3.1.4 Iscenesættelse/scenarie:

Som udgangspunkt behøver denne fase ikke at iscenesættes, men kan blot afvikles som almindelige lektioner. Læreren kan dog vælge med henblik på det scenariedidaktiske udfordrings- og konstruktionsforløb at rammesætte lektionen som et minikursus, alle kommende medarbejdere på museet skal gennemføre.

3.1.5 Faglige loops

- Introduktion til Scratch (eller CodeLab), hvis ikke eleverne allerede kender dette. Vær opmærksom på, at personlige brugerkonti i Scratch kræver forældre tilladelse indtil det fyldte 13. år. Alternativt kan muligheden for at oprette en lærerkonto anvendes, eller CodeLab på Skoletube kan bruges.
- Introduktion til, hvorledes penneværktøjet tilføjes, så det kan anvendes i programmer
- Små introduktionsopgaver. Her er det vigtigt, at læreren først gennemgår kontrolstrukturer – se præsentationen- og dernæst opgaverne, så eleverne forstår de nye svære ord, fx parametre.

3.2 Udfordrings- og konstruktionsfase

3.2.1 Varighed

Estimeret 8 lektioner á 45 minutters varighed.

3.2.2 Konkret(e) udfordring(er)

Eleverne føres igennem forløbet, enten ved hjælp af de små videoer, som er indlagt i præsentationen, eller lærerens egen rammesættelse. Eleverne skal gennem forløbet:

- Udforme skitser af Wall Drawing #273 ud fra algoritmen
- Sammenligne skitser og tilrette algoritmen, så den udtrykker lige netop deres version
- Konstruere den første væg digitalt ved hjælp af worked example
- Følge instruktioner til, hvorledes de efterfølgende vægge kan laves digitalt
- Kunne demonstrere og argumentere for deres løsning og redegøre for deres designproces

3.2.3 Programmeringsdidaktiske principper

Et "worked example" er en udførlig beskrivelse af et problem og processen frem mod en løsning. Det har vist sig at være en effektiv metode, særligt når den kombineres med "faded guidance", når man skal lære programmering. I det konkrete forløb tages udgangspunkt i et startprogram og en udførlig vejledning til konstruktionen af den første væg. Derefter anvendes den anden hjælperessource "Instruktion til væg 2-7", som kort beskriver, hvordan den første kode kan modificeres til væg 2. Derudfra skal eleverne forsøge at konstruere de efterfølgende vægge selv.

Ud over ovenstående bygger forløbet også på et andet særligt didaktisk princip i forhold til programmering, som hedder stepwise improvement. Dette princip handler om, at eleverne arbejder ud af tre akser, nemlig **konkretisering** (af delementer), **udvidelse** (frem mod det endelige produkt) og **omstrukturering** (af deres kode, efterhånden, som programmet udvides). Nybegyndere arbejder eksplorativt, når de programmerer, og princippet tilgodeser dette, da værket skal udarbejdes trinvist. Hver væg konkretiseres gennem delementer, og for hver ny væg, der laves, udvides det samlede program. Omstruktureringen sker både gennem modificering af allerede lavede koder, så de kan bruges til de nye vægge og gennem evt. omplacering af koder, som er generelle, og derfor kan ligge som egne små programmer frem for inde i de enkelte vægges programmer.

Sidst, men ikke mindst, er der udarbejdet færdige programmer til både den første væg alene og alle 7 vægge, som læreren kan vælge at dele med sine elever, hvis ovenstående er for svært. I så fald er der tale om et tredje særligt programmeringsdidaktisk princip, der kaldes "use - modify - create". Her tager eleverne udgangspunkt i et program, der virker, analyserer det og modificerer i koden, inden de laver deres egne nye elementer.

Der kan ses mere om disse tre og andre principper i de små videoer på denne side:
<https://sites.google.com/pha.dk/tek-laerer/begreber-metoder-didaktik/didaktik>

3.2.4 Faglige loops

3. og 4. lektion: Designudfordring

- **Plenum:**
Forløbet iscenesættes. Eleverne er ansat på et museum. Video 1 vises eller læreren spiller selv

rollen som museumsdirektør.
Der er fundet et gammelt brev.

- **Små grupper:**
Eleverne skal udforme skitser analogt.
- **Plenum:**
Skitser hænges op og sammenlignes

Ressourcer: Præsentation med video, kopi af brev med algoritme, elevark til skitser, som har fortrykt grid 15x15 millimeter

5. og 6. lektion: Designudfordring fortsat

- **Plenum:**
Video 2 vises. Ny opgave stilles af museumsdirektøren.
- **Små grupper:**
Eleverne skal tilrette algoritmen til at vise netop deres version af værket.
- **Plenum:**
Læreren sørger for, at eleverne "skifter roller". De er nu museumseksperter i Europa, der skal tage stilling til, hvilken version, der er bedst.
- **Små grupper:**
Grupperne udfører andre grupperes versioner ved hjælp af deres algoritmer. Algoritmerne forfines, indtil det lykkes at udføre dem helt præcist. Måske når hver gruppe kun at arbejde med en anden gruppes algoritme, men det er okay. Det er fint, hvis eleverne forholder sig til, hvilken version, de synes er bedst, men det er ikke et mål, da forløbet fortsætter scenariedidaktisk med, at "eksperterne ikke kunne blive enige".

Ressourcer: Præsentation med video, kopi af brev med algoritme, elevark til skitser, som har fortrykt grid 15x15 millimeter

7. og 8. lektion: Konstruktionsfase

- **Plenum:**
Eleverne er nu tilbage i deres oprindelige roller som ansatte på museet. Video 3 vises. Ny opgave stilles af museumsdirektøren. Eleverne skal udforme deres versioner digitalt, da eksperterne ikke kunne blive enige, og digitale versioner giver mulighed for at ændre senere.
Giv eleverne startprogrammet, efter at koderne er gennemgået (se slide 9 i præsentationen).
Startprogrammet kan findes her: <https://scratch.mit.edu/projects/313059475>
- **Små grupper:**
 - Første væg udformes i Scratch eller lignende ved hjælp af worked example
 - Dernæst udføres de efterfølgende vægge ved hjælp af instruktionsarket

Ressourcer: Præsentation med video, kopi af brev med algoritme, worked example væg 1, instruktionsark til de efterfølgende vægge.

3.2.5 Feedbackloops

Feedbackloops er indtænkt i de konkrete aktiviteter undervejs og sker gennem:

- Sammenligning af skitser i lektion 4
- Udførelse af andre gruppers algoritmer i lektion 5 og 6
- Ved ferniseringen, som er placeret i lektion 9 i outfase

Derudover kan de små videoer, som stilladserer den scenariedidaktiske ramme ses som feedbackloops.

3.3 Outrofase: Ny forståelse og nye kompetencer

I denne fase præsenterer og argumenterer eleverne for deres løsninger, de får feedback fra kammerater og reflekterer til slut over det felt, som forløbet taler ind i.

3.3.1 Varighed

Estimeret 2 lektioner á 45 minutters varighed

3.3.2 Fremlæggelse og introspektion

9. lektion: Fernisering

- **Plenum:**
Eleverne fremviser deres produkter og starter hver afsløring af værk med en lille fremlæggelse af, hvilke valg, de har taget undervejs, og hvad det har betydet for deres endelige løsning. Det anbefales, at der samtidig med det digitale værk fremvises oprindelige skitser og specificerede algoritmer, så eleverne kan bruge dem til at demonstrere, hvordan de måske har ændret undervejs fra de analoge skitser til det færdige digitale produkt.

10. lektion: Diskussion og refleksion over komplekst problemfelt: Computational kreativitet

- **Plenum:**
Video 4 vises. Eleverne møder kunstkritiker, der har været til fernisering og nu stiller en række spørgsmål, der peger ind i et komplekst problemfelt og tilhørende problemstillinger. Problemfeltet, der berøres, er computational kreativitet. På mange æstetiske felter – musik, billedkunst, litteratur – anvendes i stigende grad kunstige intelligenser til, enten selvstændigt eller i samspil med kunstneren, at skabe værker. Eleverne skal tage stilling til potentialer og implikationer ved dette og også debattere, hvad der egentlig er kunst.

Læreren stilladserer ud fra sit kendskab til klassen enten en klassedrøftelse eller en proces, hvor disse aspekter diskuteres i mindre grupper. Læreren kan evt, vise klassen eksempler fra den virkelige verden til at eksemplificere feltet:

- Aaron: Kunstig intelligens, der udfører kunst: <https://youtu.be/GrEttzMCneo>

- The Continuator: Kunstig intelligens, der udfører musik på baggrund af input:
<https://youtu.be/ynPWOMzossI>

Ressourcer: Præsentation med video

4. Perspektivering

4.1 Evaluering

Læreren er aktiv og vejledende gennem hele forløbet. Han/hun har derfor mulighed for at foretage løbende formativ evaluering af de enkelte gruppers arbejde. Læreren må meget gerne stille de mere udfordrende spørgsmål som vægter etiske overvejelser f.eks. ift. manipulation med hjemmesider. Forsøg gerne at få små diskussioner og refleksioner i gang hos eleverne.

Som evalueringsværktøj kan Læreren tjek anvendes. Lærernes tjek er en samlebetegnelse for en række spørgende og lyttende aktiviteter både skriftligt og mundtligt, hvor læreren skaffer sig indsigt i elevens færdigheder, viden eller kompetenceudvikling. Her kan vælges en dialogisk tilgang med det mål at opnå indsigt i, hvordan eleven tænker, reflekterer og argumenterer.

4.2 Progression

Forløbet bygger videre på blok 7 i 4. klasse, hvor eleverne blev introduceret kort til Scratch, og blok 5 og 6, hvor eleverne også arbejdede scenariedidaktisk. Dette forløb har fokus på samspillet mellem computationel tankegang og konstruktion af digitale artefakter, som både handler om programmeringsfærdigheder og elementer i designprocesser, og forsøger dermed at skabe en sammenhæng på tværs af 3 af fagets kompetenceområder.

4.3 Differentieringsmuligheder

Forløbet foregår hovedsagligt i grupper, som læreren kan sammensætte mest hensigtsmæssigt ud fra sit kendskab til klassen. Forløbet lægger op til en vekslen mellem plenum og gruppe aktiviteter med henblik på både at kunne lade eleverne arbejde i egne tempi, men også i gentagne loops følges og følge med i de øvrige gruppers arbejde. Særligt delen med konstruktion af det samlede digitale værk kan være svært for nogle elever at nå. Her er der udarbejdet et færdigt program, som læreren kan vælge at dele med eleverne, således at de blot skal modificere og ikke skabe det hele fra bunden.

4.4 Særlige opmærksomhedspunkter

Det er forsøgt at tilgodese følgende balanceprincipper, således at eleverne i høj grad selv får arbejdet med fabrikation og deres egne idéer, samtidig med at rammesætningen stilladserer, at idéerne ikke er frie og ustyrede og måske med øje for elevernes kompetencer umulige at udføre. Gennem den sidste lektion lægges op til, at eleverne på baggrund af deres egne fabrikationer udvider deres forståelse af teknologiers

anvendelse i virkelige kontekster.

Elevcentrerede kreative design- og læreprocesser \longleftrightarrow det rammesatte lærerstyrede undervisningsmiljø

Fabrikation af digitale artefakter \longleftrightarrow teknologiforståelse og digital myndiggørelse

5. Udvidelse af forløbet

Det er muligt at udvide forløbet med et designforløb, hvor eleverne arbejder med deres helt egne kunstværker i Scratch, fx

4-6 lektioner	Enten	Eller
	<p>Eleverne udformer deres egne generative "kunstværker" og formulerer en algoritme til en anden gruppe, som forsøger at udføre værket.</p> <p>Iterationer, indtil algoritmen er præcis nok (forskellige grupper, der afprøver).</p>	<p>Eleverne skal huske tilbage på nogle af de værktøjer, som de mødte i de første 2 lektioner, og skal udforme deres helt eget værk, som udstilles til sidst.</p> <p>Idéen skal udtrykkes som skitser + evt. pseudokode/flowcharts, inden den programmeres.</p>