

Til læreren

Formål med forløbet:

Vi kommer til at arbejde problemløsende og vil med forløbet arbejde med kompetencer inden for innovation, kreativitet og programmering.

Eleverne skal igennem forløbet arbejde med pseudokoder og mBlock (software) for at programmere mBotten.

Målgruppe og fag:

6. klasse, i valgfaget teknologiforståelse.

Forhåndskendskab:

Forløbet er lavet ud fra den antagelse, at eleven har kendskab til blokprogrammering i Scratch. Eleven skal have kendskab til, hvordan blokprogrammering overordnet fungerer og til fanerne "Hændelser" og "Styring" (mere specifikt funktioner som hvis-, gentag-, for evigt- og stopfunktionen). Vi sikrer os dog stadig, at eleverne kan huske det grundlæggende - og evt. opfriske, hvis nødvendigt.

FÆLLES MÅL

[Teknologiforståelse](#) (tiltænkt 7.-10. klasse, men passer vores forventninger til en 6. klasse):

Idé og specifikation

- Eleven kan nedbryde et problem og idéudvikling ud fra en problemskitse.
- Eleven har viden om arbejdsgange til idégenerering og problemstilling.

Realisering

- Eleven kan realisere en idé vha. programmering.

- Eleverne har viden om en iterativ udviklingsproces bestående af realisering, evaluering og forbedring.

Kodning

- Eleven kan modificere og konstruere enkle programmer.
- Eleven har viden om kodning.

LÆRINGSMÅL

TEGN PÅ LÆRING

*Læringsmål opbygget ud fra **Blooms Taksonomi**.*

*Tegn på læring udarbejdet via **Blooms Taksonomi**.*

Niveau 1 og 2:

Jeg kan genkende og forklare mBlocks kode, og bruge pseudokode som forberedelse til kodeskrivning.

Niveau 1 og 2 (orange):

- Eleven kan genkende mBlocks funktioner fra Scratch.
- Eleven kan forklare mBlocks kode (hvilken kode gør hvad?).
- Eleven kan bruge pseudokode til at forberede sig til kodeskrivning.

Niveau 3 og 4:

Jeg kan løse opgaver og lave fejlfinding i mBlock, og oversætte pseudokode direkte til programmeringskode.

Niveau 3 og 4 (grøn):

- Eleven kan bruge mBlock til at løse problemorienterede opgaver.
- Eleven kan finde fejl og rette i koden.
- Eleven kan oversætte sin pseudokode direkte til programmeringskode.

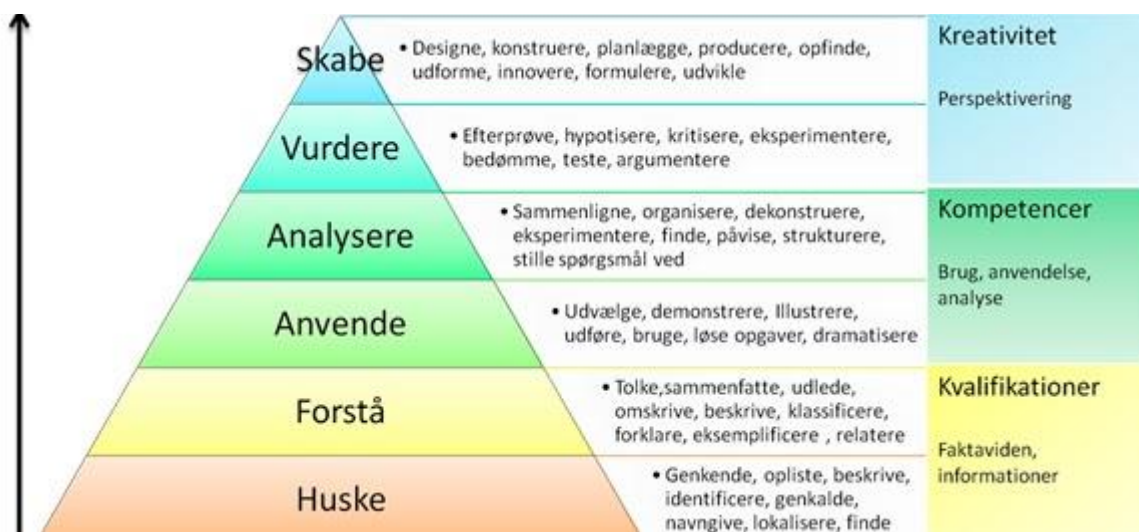
Niveau 5 og 6:

Jeg kan omskrive, forbedre, vurdere og forsvare min kode, samt selvstændigt lave pseudokode.

Niveau 5 og 6 (blå):

- Eleven kan argumentere for og bedømme, hvorfor en bestemt kode er skrevet som den er.
- Eleven kan ændre og forbedre sin kode.
- Eleven kan lave en pseudokode selvstændigt.

Til vores læringsmål og tegn på læring har vi benyttet os af Blooms modificerede taksonomi, men ikke den gængse udgave "Blooms Digitale Taksonomi" af Andrew Churches.



PROGRAMMERINGSFAGLIGE

Vi tager udgangspunkt i følgende didaktiske principper:

- Læringsforløbet dækker over flere videns- og færdighedsmål (idé og specifikation, realisering og kodning).
- Vi tager udgangspunkt i noget, der virker, dvs. eleverne stifter bekendtskab med funktionel kode, før de selv kaster sig i over arbejdet med kodning.

Dette gør vi ved at introducere eleverne til en færdigskrevet kode. Koden kan få robotten til at køre og spille en melodi. Vi vil i forbindelse med dette snakke med eleverne om koden og give dem mulighed for at danne sig tanker om, hvilken del af koden, der gør hvad. Vi vil også snakke om de individuelle kodebidder, muligheder og funktionaliteter.

- Anvendelse før produktion - eleverne arbejder med kode (pseudokode) før den egentlige kode.

I forbindelse med udvikling af deres arbejdsproces vil de starte ud med at tænke sig frem til, hvad de vil og hvordan det skal ske. Dette gøres uden for meget tanke for praktiske begrænsninger, som man så må arbejde sig uden om, såfremt det bliver nødvendigt senere i processen. Det gør vi for at understøtte den innovative proces og for at undgå, at eleverne skal føle sig begrænset i løbet af idéfasen.

- Trinvis forfinelse (stepwise improvement): Konkretisere --> Udvide --> Omstrukturere

Der er mange forskellige måder at arbejde og forbedre et stykke kode eller et projekt. I dette forløb forventer vi en stor brug af 'trial and error', da udstyret, vi arbejder med, ikke er super præcist. Så udover at skulle føre deres kodeidéer ud i livet skal eleverne også være opmærksomme på fejlkilder som hjulmodstand, uventede fejl (e.g. løse skruer, snavs og grus på gulvet, løse

ledninger på robotten) og lignende. Det betyder, at eleverne vil skulle bevæge sig frem og tilbage i de forskellige innovationsfaser.

Et eksempel på trinvis forfinelse kan både ses i det at se om robotten kører lige, for derefter at tilpasse hjulene med en værdi af 5 af gangen, teste igen, tilpasse med atter 5, osv. indtil robotten kører efter hensigt. Derudover kan trinvis forfinelse være en videreudvikling af en idé, f.eks. introduktion af lys sammen med lyd (i stedet for særskilt) eller ved at indsætte kode for at bakke (i stedet for at vende robotten 180 grader).

- Computational Thinking

Grundprincippet bag Computational Thinking er, at man udtrykker løsningen til et problem på en måde så en agent (computer) kan hjælpe med at udføre den. Det er denne idé, som pseudokode bygger på - vi forklarer, hvilke inputs vi gerne vil give computeren uden rent faktisk at bruge den egentlige kode, da dette ville tage for lang tid og dermed bremse den kreative proces.

Hvis man skal give et konkret eksempel på dette kan man se på vores andet forløb med labyrinter. Eleverne skal føre en robot igennem en labyrint, og de fleste vil ved løsning af sådan en opgave begynde at visualisere, *hvor* deres robot skal køre. Altså ikke, hvad robotten faktisk skal gøre for at komme igennem labyrinten, men hvordan selve labyrinten løses. For rent faktisk at løse opgaven må eleverne guides hen mod at bruge Computational Thinking. De kan starte med at kode robotten så den manuelt kan køre gennem labyrinten. Men hvad så, når vi giver eleverne benspænd? F.eks. kommer de til at skulle minimere mængden af manuel styring de bruger. Så vil eleverne blive tvunget til, at tænke på, hvordan robotten "tænker", så at sige. Man kan ikke fortælle mBotten "drej 270 grader mod venstre", men man kan godt fortælle den "drej venstre om med hastighed af 60 i 4 sekunder" eller "drej venstre om med hastighed af 60 så længe jeg holder knappen inde". De sidste to eksempler bruger en computational thinking logik, som kan oversættes direkte til mBottens "sprog". Denne måde at give kommandoer skal eleverne hjælpes hen imod.

- Kreativitet, Innovation, Foretagsomhed og Entreprenørskab

Disse fire begreber/stadier er benyttet til at stilladsere opgaverne i forløbet. De skal hjælpe eleverne med at få gennemarbejdet deres ideer, så de på en effektiv måde får programmeret robotten til at løse de stillede opgaver.

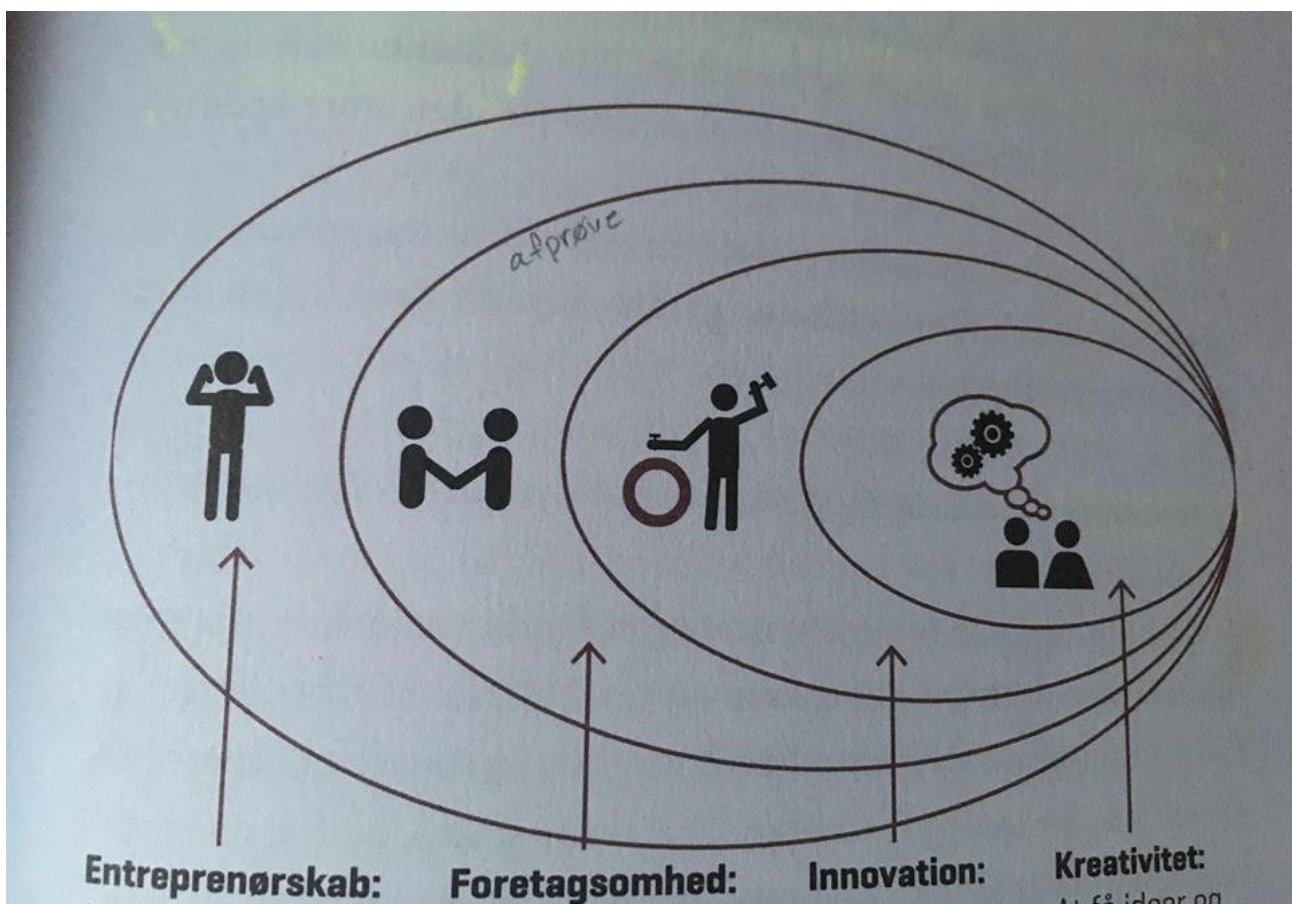
Begreberne defineres ud fra bogen "*Innovation i folkeskolen - foretagsomhed som kompetence*" af Anne Kirketerp og Karin Svennevig Hyldig, udgivet af Dansk Psykisk Forlag, på s. 15-19.

Kreativitet: At kunne skabe ideer til produkter, som er udtænkt til et bestemt formål. Lille kreativitet: løse dagligdags-problemer. Stor kreativitet: udvikle nye teknologier.

Innovation: Udvælge --> prototype --> udvikling (realisering) af de nye ideer.

Foretagsomhed: Afprøvninger af ideernes prototyper.

Entreprenørskab: Når eleverne skaber noget via egne kompetencer med udgangspunktet i noget som skal forbedres og blive en værdi for andre.



Fagfaglige anknætningspunkter/relevans

▪ **Matematik - vejledning under "programmering og matematik"**

Eleverne skal lære hvordan en computer behandler input, hvordan den afkoder dem og genererer output.

Der er tre kategorier: 1) Tænke i processer og algoritmer, 2) Digital produktion, og 3) Udvikling af abstrakt tænkning.

Du kan læse mere i [denne PDF](#) (s. 11-15) på EMUs hjemmeside.

▪ **Natur/teknologi - vejledningen under "Perspektivering i natur/teknologi"**

Omhandler forståelsen for omverden indenfor et aktuelt felt. I vores undervisningsforløb har vi taget udgangspunkt i teknologien i et mikro-perspektiv (mBots og opgaver som analogier til "den virkelige verden"), som til sidst perspektiveres til et makro-perspektiv (robotter i industrier, hverdagen, og lign.).

Du kan læse mere i [denne PDF](#) (s. 24) på EMUs hjemmeside.

▪ **Fysik/kemi - færdigheds- og vidensmål (efter 9. klassesettrin)**

Fase 3 under "Produktion og teknologi":

- Eleven kan designe og gennemføre undersøgelser vedrørende elektronisk og digital styring
- Eleven har viden om elektroniske kredsløb, simpel programmering og transmission af data

Du kan læse mere i [denne PDF](#) på EMUs hjemmeside.

▪ **Samfundsfag - fagformål**

Eleverne skal kunne arbejde med problemstillinger fra samfundet, arbejde med innovative arbejdsprocesser og arbejde sig imod et løsningsforslag. Eleverne skal også lære omkring udviklingsmuligheder og samfundsudviklingen i den globaliserede verden.

Du kan læse mere på [EMUs hjemmeside](#) under "Innovation og entreprenørskab".