

# Programmierung - leg med algoritmer.

→↑↓→

Af Klaus Eisenhardt



## Indhold

Intro: .....	1
Hour of code .....	2
Din første programmering .....	2
Programmeringskort .....	3
Algoritme.....	3
Programmering med farver .....	3
Programmer et flag .....	4
Løkker.....	5
Betingelser .....	5
Programmer et flag med løkker og betingelser .....	5
Aktiviteter .....	6
Bevægelsesopgave.....	6
Bevægelsesopgave med betinget løkke.....	6
Programmering af geometriske former .....	7
Rutealgoritme med geometrisk figur.....	7
Opgaveark .....	8
Programmeringskort .....	8
Algoritme.....	8
Programmerings sprog.....	2

## Intro:

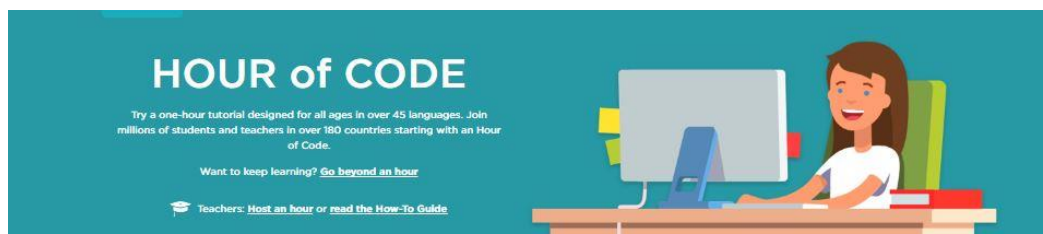
En robot, er styret vha. en computer der har brug for en køreplan/vejledning til hvad den skal gøre. Denne køreplan skal være klar og entydig dvs. den ikke skal kunne misforstås. Køreplan hedder en algoritme, du kender selv algoritmer fra bl.a. matematik, f.eks. når du skal lave regnestykket  $623+361$ , hvor algoritmen kan se sådan ud:

1. Skriv 623 på et ternet papir, med et ciffer i hver tern, efter hindanden.
2. Skriv 361 under forrige tal, så enere, tiere og hundrede står på samme række.
3. Adder enerne  $3+1$ , og skriv resultatet på tredje linje under 391
4. Adder tierne  $2+6$ , og skriv resultatet under 6-tallet.
5. Adder hundrederne  $6+3$ , og skriv resultatet under 3-tallet.
6. Summen af  $623+361$  står på linje tre.

## Hour of code

Lad eleverne stifte bekendtskab med programmering ved at lade dem vælge frit mellem spillene, på Hour of code, - begræns dog tidsforbruget ved kun at bruge et par lektioner på legen.

Klik på billedet for at komme til siden.



### Activities in your language



Moana: Wayfinding with Code  
Grades 2+ | Blocks



Minecraft Hour of Code  
Grades 2+ | Blocks



Star Wars: Building a Galaxy with Code  
Grades 2+ | Blocks, JavaScript

## Programmerings sprog

Belært af erfaring ved jeg hvor vigtigt det er fra starten af et programmeringsforløb at være skarp på programmeringssproget, altså tegnenes betydning. Endvidere er det nødvendigt på at fokusere på syntaksen, så der skrives  $\leftarrow\leftarrow\leftarrow$ , fremfor  $3\leftarrow$  eller  $3x\leftarrow$ .

På baggrund af ovenstående, har jeg brug følgende tegn og deres tilhørende betydning.

Tegn	Betydning
$\leftarrow$	Bevæg markør til venstre
$\rightarrow$	Bevæg markør til højre
$\downarrow$	Bevæg markør til ned
$\uparrow$	Bevæg markør til venstre
$\infty$	Løkke start/slut
B (efterfulgt af tal)	Betingelse gentage antal gang
S	Betyder skraver
SR	Betyder skraver Rødt
SG	Betyder skaver Grøn
SB	Betyder skraver Blå

## Din første programmering

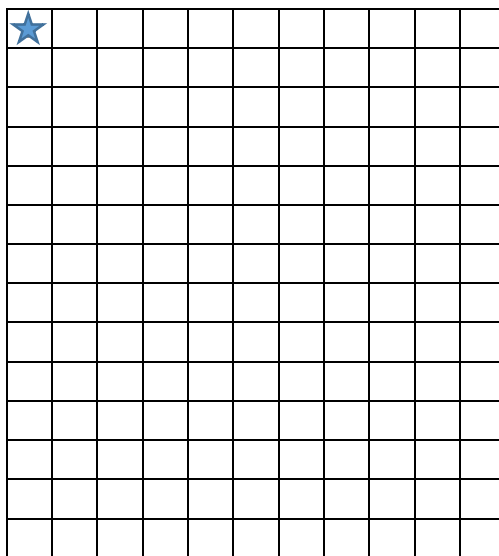
Du skal nu lave en algoritme så din samarbejdspartner kan lave en kopi af din tegning.

## Programmering - leg med algoritmer.

Gør følgende:

1. Gå sammen med din samarbejdspartner
2. Tegn et kvadrat på 15 x 15 tern, med en stjerne øverst til venstre som vist nedenfor.
3. Lav en simpel tegning ved at skraver nogle tern.
4. Brug kommandoerne ↑ for op, ↓ for ned, ← for venstre, → for højre til at lave algoritme for at tegne tegningen.
5. Brug kommandoen S for skraver til at angive hvilket felt der skal skraveres.
6. Kontroller at tegningerne er identiske.

### Programmeringskort



### Algoritme

Her kan eleven skrive algoritme der passer til opgaven


### Programmering med farver

Programmeringsproget kan nu evt. udvides til:

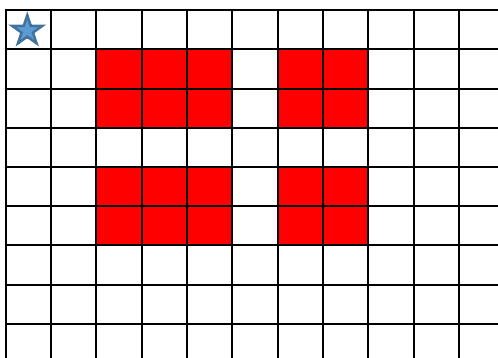
SR	SB	SG	SG
Skraver Rød	Skraver Blå	Skraver Grøn	Skraver Gul

### Programmer et flag

For at give eleverne en konkret opgave, så kan du med fordel bede dem om at lave en algoritme til tegning af et flag. Bed eleverne om at tegne kortet i programmeringskortet, og derefter lave algoritmen.



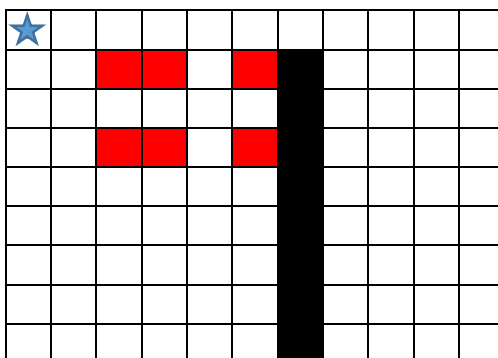
### Eksempelvis



Dannebrog med denne algoritme

→→↓SR→SR→SR→→SR→SR↓←←←←← SR→SR→SR→→SR→SR↓←←←←←↓ SR→SR→SR→→SR→SR↓←←←←←  
 SR→SR→SR→→SR→SR↓←←←←←

Og Dannebrog på flagstang



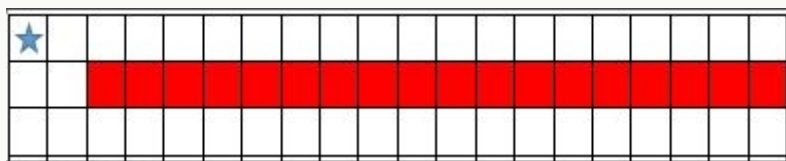
Med denne algoritme

→→↓SR→SR→→SR↓←←←←←↓SR→SR→SR→→SR→↑↑SS↓SS↓SS↓SS↓SS↓SS↓SS↓SS

## Løkker

En robot kan udføre den samme opgave igen og igen i det uendelige, til dette for at gøre algoritmen så kort som muligt bruges løkker til at gentage samme kommando i det uendelige.

Et eksempel på dette kunne være:



Med denne algoritme:

$\rightarrow \rightarrow \downarrow \infty \rightarrow SR \infty$

Hvor  $\infty$  markerer løkkens start og slut. SR betyder skraver Rød.

Robotten vil i dette tilfælde gå til højre, skraver rød, gå til højre, skraver rød, osv. Løkken er altså skraver gå til højre, skraver rød og vil fortsætte i det uendelige

## Betingelser

Betingelser er et trin i algoritmen hvor robotten skal tage stilling til et udsagn, eksempelvis:

- Gå til venstre hvis feltet er hvidt
- Stop hvis der ikke er mere plads

En betingelse kan også sættes ind foran en løkke så løkken kun gentages x antal gange, som vist nedenfor.



Med denne algoritme:

$\rightarrow \rightarrow \downarrow B4 \infty \rightarrow SR \infty$

Hvor B4 betyder at løkken gentages 4 gange.

## Programmer et flag med løkker og betingelser

Bed nu eleverne om at lave de samme flag som tidligere, dog med brug af løkker og betingelser.

Med udgangspunkt i algoritmen for Dannebrog's flaget der så således ud:

$\rightarrow \rightarrow \downarrow SR \rightarrow SR \rightarrow SR \rightarrow \rightarrow SR \rightarrow SR \downarrow \leftarrow \leftarrow \leftarrow \leftarrow SR \rightarrow SR \rightarrow SR \rightarrow \rightarrow SR \rightarrow SR \downarrow \leftarrow \leftarrow \leftarrow \leftarrow \downarrow \downarrow SR \rightarrow SR \rightarrow SR \rightarrow \rightarrow SR \rightarrow SR \downarrow \leftarrow \leftarrow \leftarrow \leftarrow$   
 $SR \rightarrow SR \rightarrow SR \rightarrow \rightarrow SR \rightarrow SR \downarrow \leftarrow \leftarrow \leftarrow \leftarrow$

Kan den forkortes til:

Dannebrog med denne algoritme

→→↓B2 ∞SR→SR→SR→→SR→SR↓←←←←←∞←←←←↓∞SR→SR→SR→→SR→SR↓←←←←←∞←←←←↓

Eller endnu kortere.

→→↓B2 ∞ B3∞SR→ ∞→→SR→SR↓B5∞←∞∞B5∞←∞↓∞ B3∞SR→ ∞→→SR→SR↓ B5∞←∞∞B5∞←∞↓

Heraf ses at en algoritme hurtig bliver kompliceret og uoverskuelig, på den anden side bliver antallet af tegn væsentligt minimeret når der bruges løkker med betingelser.

## Aktiviteter

### Bevægelsesopgave

Lad eleverne lave algoritmer over ruter fra punkter til punkter på skolen, kombinere evt. dette med et krav om at algoritmen skal omkring tredje punkt eksempelvis:

Lav en algoritme over ruten fra indgang A til C, om skolen fodboldbane

### Bevægelsesopgave med betinget løkke

Lad eleverne lave algoritmer over ruter hvor der er mulighed for at bruge løkker og betingelser. Som nedenstående, hvor eleverne skal op af flere trapper, med samme antal trin

Lav en algoritme over ruten fra indgang A til fysiklokalet. Brug løkker og betingelser i jeres programmering af ruten

Algoritmen i det aktuelle tilfælde ser således ud:

↑←↑B6∞→↑↑↑↑↑↑↑↑↑↑↑↑↑↑↑↑∞

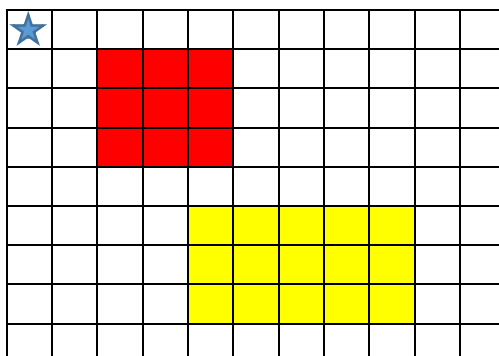
Eller endnu kortere

↑←↑B6∞→B10∞↑∞→↑↑↑∞



## Programmering af geometriske former

Når der arbejdes med geometriske former er der også inddraget programmering når der skal konstrueres figurer.



Det røde kvadrats omkreds har derfor følgende algoritme:

→→→→↓↓↓←←↑↑

Og det gule rektangel denne algoritme:

→→→→→↓↓↓←←←←↑↑↑↑

Jeg har oftest skrevet disse og lignende algoritmer op på tavlen, og spurgt klassen om hvilke geometriske form hver enkelt algoritme afbilder.

Spørg eleverne om hvordan der kan se at figurerne er hhv. kvadrater og rektangler.

### Rutealgoritme med geometrisk figur

Bevægelsesopgaverne kan også kombineres med geometriske figurer som f.eks. nedenfor.

Lav en algoritme over ruten fra indgang A til C, undervejs skal i lave et kvadrat omkring det store Lindetræ i skolegården.





Programmering - leg med algoritmer.