

Programmering og Problemløsning

1.1: Scratch

Undervisere



Jon Sparring

Image processing



Christina Lioma

Information Retrieval



Martin Elsman

Programming languages

Instruktører og mentorer

Hold	Lokale	Instruktør	Mentor
1	DIKU - 1-0-14	Pernille Horn	Emma Leschly
2	DIKU - 1-0-22	Simon Rotendahl	Layla Guelli
3	DIKU - 1-0-34	Karen Larsen	Nathalia Krøigaard Jørgensen
4	DIKU - 1-0-10	Zi Sian Peter Lim	Maja Ling Han
5	DIKU - 1-0-18	Ayman Hussein	Mathias Niebuhr Bjerregaard
6	DIKU - 1-0-30	Troels Korreman Nielsen	Mads Østerø Nørregaard
7	HCØ-Aud 6	Mikkel Sørensen	Christoffer Karlborg
8	HCØ-Aud 6	Emillie Burkal	Jacob Olesen
9	DIKU - 1-0-37	Mads Obitsø Thomsen	Jonas Flach-Jensen
10	DIKU - 1-0-26	Casper Frandsen	Magnus Joensen
11	HCØ - Aud 5	Balder Ivarsen	Ronnie Andersen
12	HCØ - Aud 5	Jan Rolandsen	Thor Steen Larsen
13	DIKU - 4-0-17	August Laursen	Matilde Broløs
14	DIKU - 4-0-17	Mads Larsen	August Clement Leve
15	DIKU - 1-0-04	Laura Boesen	-

Cami Dalsgaard

Absalon



≡ 5100-B1-2E19;Programmering og problemløsning

Home

Announcements

Assignments

Discussions

Files

People

Grades

Google Drive

Office 365

Outcomes

Modules

Collaborations

Conferences

Quizzes

Pages

Collaborations

Recent announcements

Programmering og problemløsning

Edit



Velkommen til Programmering og problemløsning

På dette kursus vil du lære at programmere efter paradigmerne: imperativ, funktions, og objektorienteret. Kurset er organiseret i moduler, som kan tilgås i menuen til venstre. Vi vil benytte F# som programmeringssprog, kommandolinjen til at afvikle programmerne, og programmerne vil blive indtastet med Emacs-editoren. De overordnede skemaer, noter m.m. findes under diverse menupunkter til venstre, og kan ligeledes tilgås ved at følge linksene nedenfor:

- [Typisk skema for 1. semester på 1. år i datalogi](#)
- [Forelæsnings- og læseplan](#)
- [Oversigt over instruktører og øvelseslokaler](#)
- [Noter, links, software m.m.](#)

Den formelle kursusbeskrivelse findes her:

- PoP@kurser.ku.dk

God fornøjelse, Jon Sparring, Kursusadministrator.

Tælleopskrifter

Metode 1

1. Første person er nummer 1
2. Nabo er nummer $n+1$

Metode 2:

1. Tæl antal personer per række vha. Metode 1
2. Sum rækkernes antal

Imperativ programming



Programmering og Problemløsning

Datalogisk Institut, Københavns Universitet

Arbejdsseddel 1 - gruppeopgave

Jon Sparring

2. september - 6. september.
Afleveringsfrist: fredag d. 6. september kl. 17:00.

Velkommen til kurset “Programmering og Problemløsning”.

Dette er kursets første *arbejdsseddel*. Arbejdssedler indeholder information om opgaver samt diverse praktisk information og perspektivering af pensum. Vi har 16 undervisningsuger, og de fleste arbejdssedler vil omhandle en undervisningsuge.

Arbejdssedlerne vil beskrive pensum, som gennemgås til forelæsningsne, øvelsesopgaver, som der vil blive arbejdet med under øvelserne, og afleveringsopgaver, som der også bliver tid til at kigge på til øvelserne. Pensum angives under “Forelæsnings- og læseplan” på Absalon og evt. tilhørende materiale finder i under Absalonpunktet “Noter, links, software m.m.”. Ca. halvdelen af afleveringsopgaverne bliver individuelle opgaver, alle andre opgaver løses i grupper.

Emnerne for denne arbejdsseddel er:

- stifte bekendtskab med imperativ programmering gennem Scratch
- stifte bekendtskab med kommandoterminalen og filstrukturer
- stifte bekendtskab med en teksteditor
- lave et spil
- lave en rapport i LaTeX
- lave den første Absalon aflevering

Opgaverne er delt i øve- og afleveringsopgaver. I denne periode skal I arbejde i grupper med jeres afleveringsopgaver. Regler for gruppe- og individuelle afleveringsopgaver er beskrevet i “Noter, links, software m.m.” → “Generel information om opgaver”.

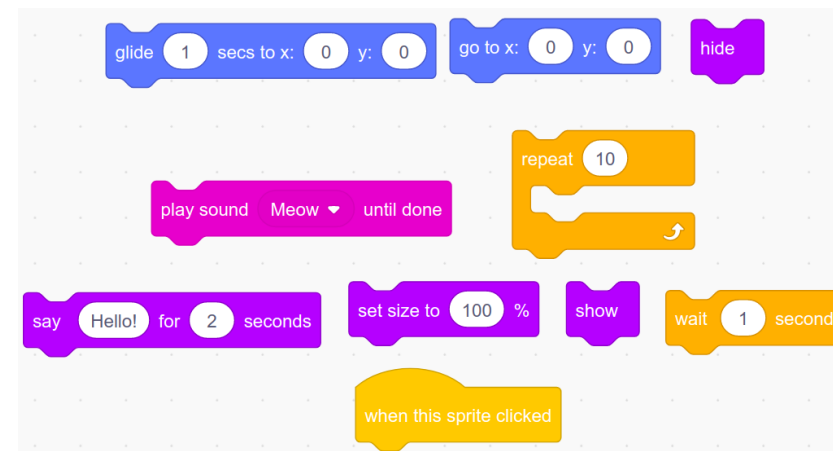


Figure 1: 10 Scratch-blokke

Øveopgaver

1ø.0 Installér Scratch, Emacs og LaTeX

1ø.1 Start Scratch, og lav et lille program, som flytter katten (eller en anden sprite) rundt på skærmen vha. glide-blokken og gentagelse.

1ø.2 Tag et eller flere skærbilleder af jeres program, mens det kører.

1ø.3 Skriv en kort rapport i LaTeX vha. Emacs og oversæt den til pdf via kommandoterminalen. Rapporten skal som minimum indeholde:

- En titel ved brug af `\maketitle`,
- et afsnit vha. `\section`,
- en eller flere figurer med skærbillederne fra jeres Scratch program med `\begin{figure}` og `\end{figure}`, og som har en figurtekst,
- en henvisning til figuren ved brug af `\label` og `\ref` strukturen.
- de danske bogstaver æ, ø, og å.

Afleveringsopgaver

1g.0 Hvad kan I lave med 10 blokke?

I Figur 1 ser I 10 Scratch blokke. Jeres opgave er at lave et sjovt program kun ved brug af disse blokke. Hver blok må bruges 0, 1 eller flere gange. Prøv at sammensætte programmet ved at tegne blokkene på papir, og skriv ned, hvad I tror programmet vil gøre. Sæt jer dernæst til computeren, og indtast jeres program. Beskriv, i hvor høj grad programmet gør, som I forventede. Vend dernæst tilbage til designfasen og forbedre evt. programmet. Til slut uploades programmet til gruppens studio i Scratch.

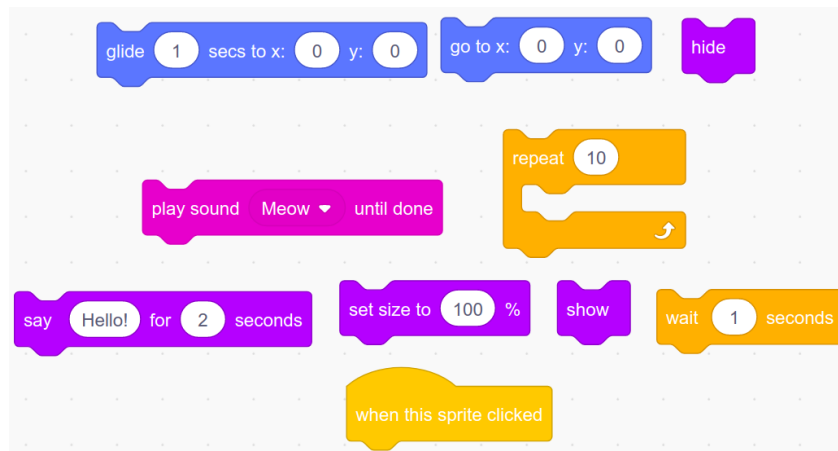


Figure 1: 10 Scratch-blokke

Øveopgaver

1ø.0 Installér Scratch, Emacs og LaTeX

1ø.1 Start Scratch, og lav et lille program, som flytter katten (eller en anden sprite) rundt på skærmen vha. glide-blokken og gentagelse.

1ø.2 Tag et eller flere skærbilleder af jeres program, mens det kører.

1ø.3 Skriv en kort rapport i LaTeX vha. Emacs og oversæt den til pdf via kommandoterminalen. Rapporten skal som minimum indeholde:

- En titel ved brug af `\maketitle`,
- et afsnit vha. `\section`,
- en eller flere figurer med skærbillederne fra jeres Scratch program med `\begin{figure}` og `\end{figure}`, og som har en figurtekst,
- en henvisning til figuren ved brug af `\label` og `\ref` strukturen.
- de danske bogstaver æ, ø, og å.

Afleveringsopgaver

1g.0 Hvad kan I lave med 10 blokke?

I Figur 1 ser I 10 Scratch blokke. Jeres opgave er at lave et sjovt program kun ved brug af disse blokke. Hver blok må bruges 0, 1 eller flere gange. Prøv at sammensætte programmet ved at tegne blokkene på papir, og skriv ned, hvad I tror programmet vil gøre. Sæt jer dernæst til computeren, og indtast jeres program. Beskriv, i hvor høj grad programmet gør, som I forventede. Vend dernæst tilbage til designfasen og forbedre evt. programmet. Til slut uploades programmet til gruppens studio i Scratch.

1g.1 Design et spil

I skal designe og implementere et spil efter eget valg. Spillet skal

- indeholde 2-5 sprites
- vare ca. 1 minut at spille
- benytte mindst 1 variabel

Det må benytte alle tilgængelige blokke i Scratch og må gerne minde om et spil I kender. Det er ikke vigtigt, at det er et grafisk eller lydæssigt prangende spil.

Start med at tale om hvad I kunne tænke jer, spillet skal omhandle. Skitser på papir, hvordan game-playet, skal forløbe. Skitser derefter på papir hvordan det kunne implementeres i Scratch. Indtast programmet på computeren og afprøv, om spillet gør, som I forventer. Vend evt. tilbage til designfasen og forbedre spillet.

Afleveringen skal bestå af

- upload af de udviklede programmer i klassens studio på Scratch' hjemmeside,
- en zip-fil og
- en rapport i pdf-format.

Rapporten skal dokumentere jeres process og beskrive jeres programmer. Den skal skrives i LaTeX med et passende antal billeder, illustrationer og henvise til jeres programmer. LaTeX koden (.tex) inklusiv billeder skal organiseres i en mappe kaldet tex, således at pdf-filen kan laves ved kørsel af `pdflatex` kommandoen i mappen. Mappen skal zippes (.zip) og afleveres sammen med den oversatte pdf fil (.pdf) som 2 filer i Absalon.

God fornøjelse.

Hvem er I?

<https://tinyurl.com/yykjc8oh>

Online programming...

- Hello World
- Count to 9

