**Programmering i NetLogo**

**Den store skildpaddevandring**

**1**. Få programmet til at skrive "Hej verden, jeg hedder <dit navn>!". Dette skal ske når du trykker på en knap som du kalder setup og som udfører underrutinen setup. I det følgende står < … > for noget tekst som du finder på: i stedet for <navn> skriver du dit navn, i stedet for <kode> skriver du noget kode osv. Du får brug for

to setup  
 clear-all  
 <kode>  
end  
show "Hej …"  
en knap som du kalder setup og som udfører underrutinen setup når der trykkes på knappen

Bemærk at "to" betyder "at" på dansk. Den første linje kan altså læses som "at sætte (programmet) op". To fortæller NetLogo at her starter der en underrutine som så senere afsluttes med end. Det er altid en god ide at starte setup underrutinen med clear-all, som nulstiller alt.Der sættes anførselstegn omkring tekst. En knap hedder button på engelsk.

**Kommentarer**. Computeren udfører alle de linjer du skriver uden undtagelse. Men af og til er det nyttigt at computeren springer linjer over. Det kan f.eks. være en god ide at skrive kommentarer i programmet:

; i næste linje skriver computeren mit navn  
show "Hej …"

En linje der starter med et semikolon springes over. Computeren udfører altså ikke kommentaren og ved heller ikke hvad den betyder. Dette kan også bruges til at lede efter fejl:

; sjow "Hej …"

Ved at udkommentere en eller flere mistænkelige linjer kan man få programmet godkendt ved tryk på check. Man ved så at fejlen må være et eller andet sted i de udkommenterede linjer. I eksemplet skal sjow rettes til show og semikolonet kan så fjernes igen.

Følgende skal også ske når du trykker på setup:

**2**. Få programmet til at farve skærmen rød. Du får brug for

ask patches [  
 <kode>  
]  
set pcolor red

Bemærk at felter hedder patches på engelsk. Der sættes et p foran color for at angive farven for et felt.

**3**. Få programmet til at tegne en hvid x-akse og en hvid y-akse. Du får brug for

if (pycor = 0) [set pcolor white]

Tilsvarende for y-aksen.  
Der sættes et p foran ycor for at angive y-koordinaten for et felt. Under settings kan man se at vinduet som udgangspunkt består af felter hvis x-koordinater ligger i intervallet [-16, 16] og y-koordinater i [-16, 16]. Feltet i midten af vinduet har altså koordinaterne (0, 0). Der er 332 = 1089 felter i alt.

**4**. Få programmet til at skrive "x" på x-aksen og "y" på y-aksen med blå skrift. Du får brug for

if (pxcor = 16 and pycor = 0) [  
 <kode>  
]  
set plabel-color blue  
set plabel "x"

Og tilsvarende for y-aksen.  
Bemærk at en mærkat hedder label på engelsk. Du kan gøre bogstaverne større ved at ændre på "font size" under settings.

Vi forlader nu felterne (patches) for at kigge på skildpadder (turtles).

**5**. Skriv linjerne

create-turtles 8  
ask turtles [set size 1.5]

lige efter clear-all kommandoen i setup underrutinen. Dette betyder at  
(i) der laves 8 skildpadder i koordinatsystemets centrum med tilfældig farve og orientering  
(ii) skildpaddestørrelsen sættes til 1.5.

**6**. Få programmet til at placere den første skildpadde i (4, 0). Den skal være grøn og pege mod øst. Øst er 90° fra nord i negativ omløbsretning (med urets omløbsretning). Du får brug for  
ask turtle 0 [  
 <kode>  
]  
set xcor 4  
set ycor 0  
set color green  
set heading 90

Bemærk at den første af de 8 skildpadder har nummer 0 og den sidste nummer 7!

**7**. Få programmet til at placere de syv andre skildpadder i

(4, 4) med farve gul og orientering nord-øst  
(0, 4) med farve grøn og orientering nord  
(-4, 4) med farve gul og orientering nord-vest  
(-4, 0) med farve grøn og orientering vest  
(-4, -4) med farve gul og orientering syd-vest  
(0, -4) med farve grøn og orientering syd  
(4, -4) med farve gul og orientering syd-øst

Vi vil nu lave en underrutine go der får skildpadderne til at flytte sig. For at de ikke kommer til at gå igennem vinduets kanter skal du under settings fjerne de to flueben ved "World wraps horizontally" og "World wraps vertically". To wrap around betyder at vikle om, så ved at fjerne fluebenene forhindrer du skildpadderne i at forsvinde ved en kant og dukke op ved den modsatte kant.

**8**. Få programmet til at flytte de otte skildpadder et skridt fremad. Dette skal ske når du trykker på en knap som du kalder go og som udfører underrutinen go. Du får brug for

to go  
 <kode>  
end  
ask turtles [  
 <kode>  
]  
forward 1

**9**. Få programmet til at flytte de grønne skildpadder et skridt til højre og de gule et skridt til venstre når du trykker på go. Du får brug for

if (color = green) [set heading 90]

Og tilsvarende for gul.  
Efter en del tryk på go vil de grønne skildpadder være samlet til højre og de gule til venstre.

**10**. Få programmet til at samle de grønne skildpadder foroven og de gule forneden.

**11**. Få programmet til at samle de grønne skildpadder i øverste højre hjørne og de gule i nederste venstre hjørne. Du får brug for

if (color = green and xcor > 15.5) [set heading 0]  
if (color = green and ycor > 15.5) [set heading 90]

Og tilsvarende for gul.

**12**. Få programmet til at samle skildpadderne uden en masse tryk på go. Du for brug for

en ny knap som du også kalder go og som udfører underrutinen go hele tiden når der trykkes på knappen,  
 dvs at i knappens indstillinger skal der sættes flueben ved forever  
wait 0.1 som sidste linje i underrutinen go

Bemærk at wait 0.1 stopper programmet i 0,1 sekunder så skildpadderne flytter sig tilpas langsomt. Du kan stoppe den gentagne udførsel af underrutinen go ved at trykke på go (forever) igen.

**13**. Få programmet til ændre skildpaddernes skridtlængde når du bevæger en skyderknap mens programmer kører. Du får brug for

en skyder som du kalder skridtlængde  
i skyderens indstillinger sætter du minimum til 0, increment til 0.1, maximum til 1 og value til 0.5  
i underrutinen go bruger du den nye variabel skridtlængde der hvor du før skrev 1 i forward linjen.

Bemærk at en skyder hedder slider på engelsk.

Vi forlader nu skildpadderne (turtles) for at kigge på hvordan vi laver lister af tal. Det følgende laves i setup.

**14**. Få programmet til at lave en liste med tallene 0, 1 og 2 som udskrives. Du får brug for

let liste (n-values 3 [i -> i])  
foreach liste [tal ->  
 show tal  
]

Her laver (n-values 3 [i -> i]) en liste der består af tre tal: {0, 1, 2}. Det første i er det løbende indeks som løber fra 0 til 2 med skridtlængde 1. Dette andet i er outputtet til listen. Her er der ingen forskel, så tallene 0, 1 og 2 sendes uændret til listen. Den dannede liste kaldes for liste til senere brug. Kommandoen foreach gennemløber listen liste element for element som outputtes til variablen tal.

**15**. Få programmet til at lave en liste med tallene -1, 0 og 1 som udskrives. Du får brug for

set liste (n-values 3 [i -> i - 1])

Bemærk at man bruger let første gang og set de følgende gange man definerer en variabel (her liste). Hvis du ikke kan huske om variablen allerede er defineret, så start med let, og hvis dette ikke virker, så brug set.

**16**. Få programmet til at udskrive tre tilfældige decimaltal mellem 0 og 10. Du får brug for

set liste (n-values 3 [i -> (random-float 10)])

**17**. Få programmet til skrive hvem af Kaj og Andrea der tager opvasken de første tre dage i næste uge. Hver dag har Kaj og Andrea begge en sandsynlighed på 0,5 for at blive valgt. Du får brug for

set liste (list "mandag" "tirsdag" "onsdag")  
foreach liste [dag ->  
 <kode>  
]  
let tal (random-float 1)  
let navn (ifelse-value (tal < 0.5) ["Kaj"] ["Andrea"])  
show (word "På " dag " tager " navn " opvasken")

Bemærk at kommandoen list laver en liste af de elementer der følger. Kommandoen ifelse-value outputter Kaj til navn hvis det tilfældige decimaltal tal mellem 0 og 1 er mindre end 0,5 og ellers outputtes Andrea. Til sidst sætter word fem tekststrenge sammen til en sætning som udskrives.

**18**. Få programmet til at lave 100 skildpadder der placeres tilfældigt og med farverne grøn eller gul når du trykker på setup. Du får brug for

foreach liste [i ->  
 <kode>  
]  
let x (random-float 32) - 16  
let y (random-float 32) - 16  
let tal (random-float 1)  
let farve (ifelse-value (tal < 0.5) [green] [yellow])  
ask turtle <nummer> [  
 <kode>  
]

Her skal liste bestå af de 100 skildpaddenumre {0, 1, 2, …, 99}.  
Tryk på go (forever) så skildpadderne vandrer hjem til Grøn-land og Gul-land!