

# ÆNDRINGERNE I SIMULERINGSKODEN

I artikelsamlingen "SGoleudviklingsprojekt 2015-2018" var der ikke plads til at vise de ændringer, der skal foretages i koden af simuleringen i de forskellige aktiviteter. Disse er vist nedenfor.

## Aktivitet 1c og 2b

I aktivitet 1c skal atomkernernes udseende (form og farve) ændres. I aktivitet 2b skal simuleringens startværdier ændres. For at løse disse opgaver, må eleverne analysere koden og finde de steder, hvor kernernes farve og form indstilles (markeret med grønt i figuren) og de steder, hvor startværdierne indstilles (markeret med blå).

```
breed [N1s N1]
breed [N2s N2]

globals [
  Sandsynlighed-for-henfald-N1
  Startantal-N1
]

to setup
  clear-all

  set-default-shape N1s "circle"
  set-default-shape N2s "circle"

  set Sandsynlighed-for-henfald-N1 3
  set Startantal-N1 8000

  create-N1s Startantal-N1 [
    set color cyan
    move-to one-of patches with [ not any? Turtles-here ]
  ]

  reset-ticks
end

to go
  if not any? N1s [ stop ]
  kør-henfald1
  tick
end

to kør-henfald1
  ask N1s [
    if random-float 100.0 < Sandsynlighed-for-henfald-N1 [
      set breed N2s
      set color red
    ]
  ]
end
```

 Ændring af kernernes udseende  
 Ændring af simuleringens værdier

## Aktivitet 3c

I aktivitet 3c skal eleverne vha. ændringer i simuleringværktøjet undersøge, hvordan aktiviteten fra en radioaktiv prøve, der hele tiden får tilført nye radioaktive kerner med en konstant rate, udvikler sig med tiden. Det gøres ved at studere kommentarerne i koden (under Code-fanen) og arbejde med rutediagrammerne over algoritmerne i simuleringen (se særskilt pdf-dokument). Eleverne skal:

1. *Finde ud af hvordan man tilføjer radioaktive kerner (N1-kerner).* Da der allerede er N1-kerner i simuleringen til at begynde med, må der være et sted i koden, hvor de tilføjes. Dette sted skal findes og kodelinjen skal kopieres.
2. *Finde ud af hvor i algoritmen denne kodelinje skal tilføjes.* Det kan gøres ved at studere koden, men det nemmeste er nok at bruge rutediagrammerne og markere dér, hvor der skal tilføjes nye N1-kerner. Dernæst findes dette sted i koden, og det kopierede kodelinje sættes ind.
3. *Rette det indsatte kodelinje til,* så der tilføres netop 100 kerner per år.

Nedenfor er vist en elevs løsning af opgaven med mine markeringer/kommentarer.

**Oprindelig kode (udleveret):**

```

breed [N1s N1]
breed [N2s N2]

globals [
  Sandsynlighed-for-henfald-N1
  Startantal-N1
]

to setup
  clear-all

  set-default-shape N1s "circle"
  set-default-shape N2s "circle"

  set Sandsynlighed-for-henfald-N1 3
  set Startantal-N1 8000

  create-N1s Startantal-N1 [
    set color cyan
    move-to one-of patches with [ not any? Turtles-here ]
  ]

  reset-ticks
end

to go
  if not any? N1s [ stop ]
  kør-henfald1
  tick
end

to kør-henfald1
  ask N1s [
    if random-float 100.0 < Sandsynlighed-for-henfald-N1 [
      set breed N2s
      set color red
    ]
  ]
end

```

Kode tilføjet (kopieret og rettet til fra andet sted)  
Kode rettet til

**Ny kode (lavet af elev):**

```

breed [N1s N1]
breed [N2s N2]

globals [
  Sandsynlighed-for-henfald-N1
  Startantal-N1
]

to setup
  clear-all

  set-default-shape N1s "circle"
  set-default-shape N2s "circle"

  set Sandsynlighed-for-henfald-N1 3
  set Startantal-N1 8000

  create-N1s Startantal-N1 [
    set color cyan
    move-to one-of patches with [ not any? Turtles-here ]
  ]

  reset-ticks
end

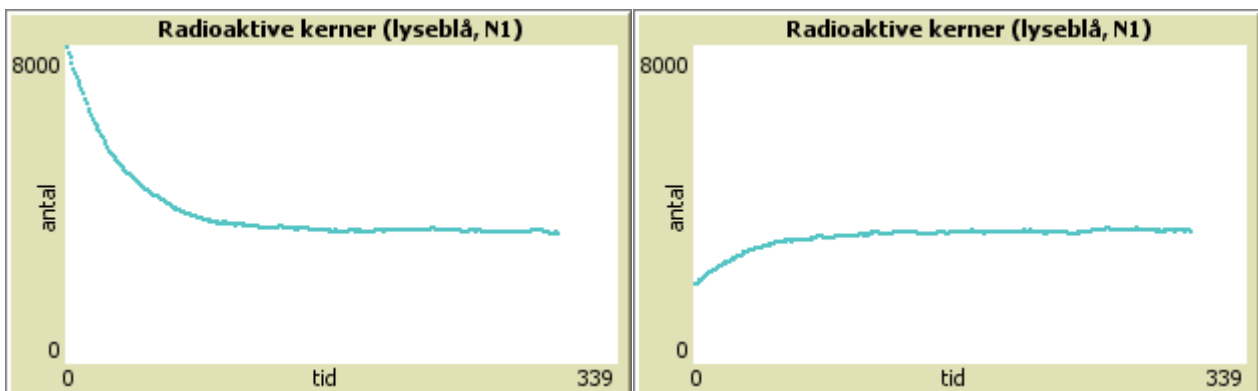
to go
  if not any? N1s [ stop ]
  kør-henfald1
  create-N1s 100 [
    set color cyan
    move-to one-of patches with [ not any? Turtles-here ]
  ]
  tick
end

to kør-henfald1
  ask N1s [
    if random-float 100.0 < Sandsynlighed-for-henfald-N1 [
      die
      ;; set breed N2s
      ;; set color red
    ]
  ]
end

```

Kopieret hertil og rettet til  
 Ændret så simuleringen ikke "dør"

Ved at ændre på startværdierne i deres nye simuleringsværktøj, kan eleverne se, at antallet af 14C-kerner i en organisme ender på et konstant niveau, der afhænger af halveringstiden for 14C og tilføjelsesraten, men ikke af startantallet:



#### Aktivitet 4b

I aktivitet 4b skal eleverne ændre i simuleringsværktøjet, så det kan anvendes til at simulere end henfaldsproces, hvor datterkernerne fra det første henfald er radioaktive og henfalder igen. De skal bruge det nye simuleringsværktøj til at undersøge, hvordan aktiviteten af det midterste stof ændrer sig med tiden. Eleverne skal igennem mange af de samme analyser og overvejelser som i aktivitet 3c - dog er de nødvendige ændringer i koden mere omfattende. I denne aktivitet varierede elevernes løsninger (kode) en del, men deres simuleringer viste alle det rigtige. Nedenfor er vist en elevs løsning (kode + interface med resultater).

### Oprindelig kode (udleveret):

```
breed [N1s N1]
breed [N2s N2]

globals [
  Sandsynlighed-for-henfald-N1
  Startantal-N1
]

to setup
  clear-all

  set-default-shape N1s "circle"
  set-default-shape N2s "circle"

  set Sandsynlighed-for-henfald-N1 3
  set Startantal-N1 8000

  create-N1s Startantal-N1 [
    set color cyan
    move-to one-of patches with [ not any? Turtles-here ]
  ]

  reset-ticks
end

to go
  if not any? N1s [ stop ]
  kør-henfald1
  tick
end

to kør-henfald1
  ask N1s [
    if random-float 100.0 < Sandsynlighed-for-henfald-N1 [
      set breed N2s
      set color red
    ]
  ]
end
```

Kode tilføjet (kopieret og rettet til fra andet sted)

Kode rettet til

### Ny kode (lavet af elev):

```
breed [N1s N1]
breed [N2s N2]
breed [N3s N3]

globals [
  Sandsynlighed-for-henfald-N1
  Startantal-N1
  Sandsynlighed-for-henfald-N2
]

to setup
  clear-all

  set-default-shape N1s "circle"
  set-default-shape N2s "circle"
  set-default-shape N3s "circle"

  set Sandsynlighed-for-henfald-N1 55.45
  set Sandsynlighed-for-henfald-N2 8.664
  set Startantal-N1 8000

  create-N1s Startantal-N1
  [set color cyan
  move-to one-of patches with [ not any? Turtles-here ]
]

  reset-ticks
end

to go
  if not any? N1s [ if not any? N2s [ stop ] ]
  kør-henfald1
  kør-henfald2
  tick
end

to kør-henfald1
  ask N1s [
    if random-float 100.0 < Sandsynlighed-for-henfald-N1 [
      set breed N2s
      set color magenta
    ]
  ]
end

to kør-henfald2
  ask N2s [
    if random-float 100.0 < Sandsynlighed-for-henfald-N2 [
      set breed N3s
      set color yellow
    ]
  ]
end
```

Ændret så simuleringen stopper, når der hverken er N1- eller N2-kerner tilbage

