

02101 Indledende Programmering 2021

Hjemmeopgave 1

Daniel Brasholt s214675
Jonathan Svarer Johansen s214671
Johannes Bjørn Nielsen s214682

Oktober 2021

Arbejdsdeling

Vi har valgt at dele hovedansvaret for hver opgave op; Jonathan har lavet opgave 1, Johannes opgave 2 og Daniel opgave 3.a og 3.b. Vi har dog hele tiden kigget hinandens kode igennem for at sørge for, at det bliver skrevet så homogent som muligt og for at fjerne eventuelle fejl i programmerne.

Rapport

3.a

I opgave 3.a ligger al koden i klassen `PredatorPrey`. Under denne er der følgende metoder:

- *runSimulation*: Hovedmetoden i programmet. Tjekker inputparametre og indeholder løkken, der bevæger Prey og Predator. Tjekker også, om Prey er blevet fanget.
- *checkInput*: Kaldes af *runSimulation* og tjekker, om input er indenfor grænserne.
- *randomInS*: Laver et tilfældigt tal, som bruges til at flytte Prey.
- *movePrey*: Flytter Prey med *randomInS* og *isInBounds()*.
- *isInBounds*: Tjekker, om et punkt ligger indenfor brætstørrelsen specificeret i input *n*.
- *movePredator*: Flytter Predator så tæt på Prey som muligt.
- *initStartPoint*: Returnerer et punkt med tilfældige koordinater indenfor *n*.
- *printPoints*: Printer punkterne i formatet givet i opgaven. - *isCaught*: Tjekker, om Prey og Predator ligger det samme sted og at Prey derfor er fanget.

runSimulation tjekker først, om input er indenfor grænserne med *checkInput*. Er parametrene i orden, laves to tilfældige punkter for Prey og Predator med *initStartPoint*. Disse printes med *printPoints* og løkken går i gang. Først tjekkes det, om Prey er blevet fanget med *isCaught*, hvorefter først Prey og dernæst

Predator flyttes med *movePrey* og *movePredator*. Til sidst i løkken printes punkterne igen. *movePrey* bruger funktionen *randomInS* samt *isInBounds* for at sørge for, at punktet ikke undslipper grænsen sat ved parameter *n*.

3.b

Da opgave 3.b og 3.a ligner hinanden meget i funktionalitet, kan de fleste funktioner blot genbruges. De metoder, der skal laves defineres igen, er *runSimulation*, *checkInput* og *movePrey* (som er omdøbt *movePreyWithTeleport*). Disse skal defineres, da

- *runSimulation* indeholder hovedløkken og derfor skal være defineret.
- *checkInput* skal tjekke efter en ny grænseværdi for *s*.
- *movePreyWithTeleport*, da Prey skal kunne teleportere, hvis begge koordinatsæt er divisible med *s*.

Udover disse funktioner, arver *PredatorPrayTeleport*-klassen blot metoderne fra *PredatorPray*, hvorfor begge disse to klasser skal være til stede, for at *PredatorPrayTeleport* kan kompilere.

Den overordnede funktionalitet og kørsel er da ens for *PredatorPray* og *PredatorPrayTeleport*; de samme metoder kalder hinanden og i den samme rækkefølge.