

Learning to Program with F#
Exercises
Department of Computer Science
University of Copenhagen

Jon Sparring, Martin Elsmann, Torben Mogensen, Christina Lioma

October 21, 2022

0.1 Wolves and mooses

0.1.1 Exercise(s)

0.1.1.1: I det følgende skal der simuleres et lukket miljø med ulve og elge. Simuleringen skal benytte følgende regler:

- (a) Et miljø består af $n \times n$ felter.
- (b) Alle levende dyr har en koordinat i miljøet, og der kan højst være et dyr per felt. Når et dyr dør, fjernes det fra miljøet. Hvis et dyr fødes, tilføjes det i et tomt felt. Ved simuleringens begyndelse skal der være u ulve og e elge som placeret tilfældigt i tomme felter.
- (c) Miljøet opdateres i tidsenheder, som kaldes tiks, og simuleringen udføres T tiks. Indenfor et tik kan dyrene gøre et af følgende: Flytte sig, formere sig, og for ulvenes vedkommende spise en elg. Kun et dyr handler ad gangen og rækkefølgen er tilfældig.
- (d) Dyr kan flytte sig et felt per tik til et af de 8 nabofelter, som er tomme.
- (e) Alle dyr har en artsspecifik formeringstid f angivet i antal tiks, og som tæller ned. Når formeringstiden når nul (for et levende dyr), og der er et tomt nabofelt, så fødes der et nyt dyr af samme type ved at det nye dyr tilføres i et tomt nabofelt. Moderdyrets formeringstid sættes til startværdien, hhv. f_{elg} og f_{ulv} .
- (f) Ulve har en sulttid s angivet i antal tiks, og som tæller ned. Hvis sulttiden når nul, så dør ulven, og den fjernes fra miljøet.
- (g) Ulve kan spise elge. Hvis der er en elg i et nabofelt vil ulven spise elgen, elgen fjernes fra miljøet, ulven flytter til elgens felt, og ulvens sulttid sættes til startværdien, s .
- (h) I hvert tik reduceres alle formerings- og sulttællere for levende dyr med 1.

Lav et program, som kan simulere dyrene som beskrevet ovenfor og skrive en rapport. Til opgaven udleveres følgende kildefiler:

`animalsSmall.fsi`, `animalsSmall.fs`, og `testAnimalsSmall.fs`.

Opgaven er at tage udgangspunkt i disse filer og programmere følgende regler:

- (a) Der skal laves et bibliotek som implementerer klasser for miljø, ulve og elge. Det er ikke et krav at der bruges nedarvning.
- (b) Man skal kunne starte simuleringen med forskellige værdier af T , n , u , e , f_{elg} , f_{ulv} og s .
- (c) Der skal laves en white-box test af biblioteket.
- (d) Der skal laves en applikation, som kører en simulering, og tidsserien over antallet af dyr per tik skal gemmes i en fil. Filnavn og parametrene T , n , e , f_{elg} , u , f_{ulv} og s skal angives som argumenter til det oversatte program fra komandolinjen. Eksempelvis kunne:

```
mono experimentWAnimals.exe 40 test.txt 10 30 10 2 10 4
```

starte et eksperiment med $T = 40$, $n = 10$, $e = 30$, $f_{\text{elg}} = 10$, $u = 2$, $f_{\text{ulv}} = 10$ og $s = 4$ og hvor tidsserien skrives til filen `test.txt`.
- (e) Der skal laves et antal eksperimenter, hvor simuleringen køres med forskellige værdier af simuleringens parametre. For hvert eksperiment skal der laves en graf (ikke nødvendigvis i F#), der viser antallet af ulve og elge over tid.
- (f) Koden skal kommenteres ved brug af F# kommentarstandarden.

Kravene til rapporten er:

- (g) Rapporten skal skrives i \LaTeX og tage udgangspunkt i `rapport.tex` skabelonen
- (h) Rapporten skal som minimum indeholde afsnittene Introduktion, Problemanalyse og design, Programbeskrivelse, Afprøvning, Eksperiment og Konklusion. Som bilag skal I vedlægge afsnittene Brugervejledning og Programtekst.
- (i) Eksperimentafsnittet skal kort diskutere hvert eksperiments udfald.
- (j) Rapporten minus bilag må maksimalt være på 10 A4 sider alt inklusivt.