

# Øving 5

INF620 - Høstsemesteret 2021

Øvingsoppgavene er ikke obligatoriske, men vi anbefaler likevel at du gjør de og leverer de innen fristen — Den eneste måten å lære å programmere på er ved å programmere. Ved å gjøre oppgavene får du også testet deg selv og sjekket at du forstår begrepene. Du skal levere én zip-fil, **oving5.zip**, som inneholder de 2 filene **oppg1.py**–**oppg2.py**. For å komprimere en eller flere filer til en zip-fil høyreklikker du filene (i dette tilfellet **oppg1.py**–**oppg2.py**) i maskinens filnavigasjonsprogram og velger **Komprimer** eller **Send til → Komprimert mappe**. Frist: Torsdag 21. oktober kl 23:59

## 1 Nedbørsobservasjoner

Dette er en oppgave som gir trening i bruk av *endimensjonale* lister. Svar leveres på fil med navn **oppg1.py**

- a) Skriv en funksjon **lag\_liste(n, sanns, maks)** som returnerer en liste med nedbørsverdier for **n** dager. Sannsynligheten for at en dag er en nedbørsdag er gitt med parameteren **sanns**. På nedbørsdager er nedbøren et tilfeldig flyttall på intervallet fra og med 0.0 til verdien på parameteren **maks**. På andre dager er nedbøren lik 0.0. Eksempel på kjøring av funksjonen fra konsollen i Spyder:

```
In [1]: lag_liste(10, 0.4, 8.0)
Out[1]:
[0.0,
 0.8881669814300528,
 7.639636352747872,
 0.0,
 5.354892034229507,
 1.7660155302203941,
 0.0,
 0.0,
 0.0,
 0.0]
```

- b) Skriv en funksjon **gjennomsnitt(nedbør)** som returnerer en liste med gjennomsnittsnedbør fra og med første dag dekket av listen **nedbør**. For hver dag **i** skal den returnerte listen inneholde gjennomsnittet av nedbør f.o.m. dag 0 t.o.m. dag **i**. Eksempel på kjøring av funksjonen fra konsollen i Spyder:

```

In [3]: gjennomsnitt(
        [0.0, 3.2, 2.8, 0.0, 4.2, 4.5, 1.1, 0.0, 0.9, 0.7,
         3.1, 0.0, 0.0, 0.0, 2.2, 2.3, 2.7, 0.0, 0.0])
Out[3]: [0.0,
        1.6,
        2.0,
        1.5,
        2.04,
        2.4499999999999997,
        2.257142857142857,
        1.9749999999999999,
        1.8555555555555554,
        1.7399999999999998,
        1.8636363636363635,
        1.7083333333333333,
        1.5769230769230769,
        1.4642857142857142,
        1.5133333333333332,
        1.5625,
        1.6294117647058823,
        1.5388888888888888,
        1.4578947368421051]

```

- c) Skriv en funksjon `økning(nedbør)` som tar en nedbørsliste som parameter, og som returnerer en liste med nedbørsøkninger fra dagen før. Vi antar at det ikke var nedbør siste dag før observasjonene startet. Eksempel på kjøring av funksjonen fra konsollen i Spyder:

```

In [4]: økning(
        [0.0, 3.2, 2.8, 0.0, 4.2, 4.5, 1.1, 0.0, 0.9, 0.7,
         3.1, 0.0, 0.0, 0.0, 2.2, 2.3, 2.7, 0.0, 0.0])
Out[4]:
[0.0,
 3.2,
-0.40000000000000036,
-2.8,
 4.2,
 0.29999999999999998,
-3.4,
-1.1,
 0.9,
-0.20000000000000007,
 2.4000000000000004,
-3.1,
 0.0,
 0.0,
 2.2,
 0.099999999999999964,
 0.40000000000000036,
-2.7,
 0.0]

```

- d) Denne oppgaven er kun for de som er interesserte i å gå utenfor pensum. Skriv en funksjon `plott(nedbør)` som gjør bruk av funksjonene `gjennomsnitt` og `økning`. Deretter skal funksjonen plote nedbørsverdiene i listen `nedbør` sammen med tilhørende gjennomsnittsverdier (se oppg. 1b) og økninger (se oppg. 1c). For plotting av tre lister, her kalt `liste1`, `liste2` og `liste3`, kan du bruke:

```

from matplotlib import pyplot
pyplot.plot(liste1)
pyplot.plot(liste2)
pyplot.plot(liste3)
pyplot.show()

```

## 2 Kryssord

Svar leveres på fil med navn `oppg2.py`. Dette er en oppgave som gir trening i bruk av *todimensjonale* lister.

Vi skal skrive kode som skal hjelpe en kryssordleverandør å lage rutemønstre. Et kryssord betraktes som et rektangel delt opp i *åpne* og *lukkede* ruter. I koden som skal utvikles i denne oppgaven representeres rutemønstret med en todimensjonal liste bestående av karakterer. Hvilke ruter som skal være hhv. åpne og lukkede bestemmes delvis av enkle regler, og delvis av tilfeldigheter.

Som symbol på hhv. en åpen og en lukket rute brukes karakterene 'O' (stor 'o') og 'X' (stor 'x'). Derfor kan starten av filen `oppg2.py` gjerne inneholde kodelinjene:

```
ÅPEN = 'O'  
LUKKET = 'X'
```

- a) Skriv en funksjon `skriv(kryssord)` som får en todimensjonal liste svarende til et kryssordmønster som parameter, og som skriver mønsteret ryddig ut i konsollen. Eksempel på utskrift når funksjonen kjøres fra konsollen i Spyder:

```
In [5]: skriv(  
[[ 'O', 'X', 'O', 'X', 'O'],  
 [ 'X', 'O', 'O', 'O', 'O'],  
 [ 'O', 'O', 'X', 'X', 'X'],  
 [ 'X', 'O', 'X', 'O', 'O']])  
OXOXO  
XOOOO  
OOXXX  
XOXOO
```

- b) Rutene i kryssordet identifiseres av to indekser. Første indeks gir radnummeret (0 er øverste rad, antall rader minus en er nederste rad). Andre indeks gir kolonnennummeret (0 er kolonnen lengst til venstre, antall kolonner minus en er kolonnen lengst til høyre). Skriv en funksjon `har_nabo(kryssord, rad, kol)` som tar et rutemønster som første parameter, og indeksene til en rute i kryssordet som de to siste parametrene. Funksjonen skal returnere `True` dersom ruten i rad `rad` og kolonne `kol` har en *åpen* naborute (over, under, til venstre eller til høyre). I motsatt tilfelle skal `False` returneres. Eksempler på kjøring av funksjonen fra konsollen i Spyder:

```
In [6]: har_nabo(  
[[ 'O', 'X', 'O', 'X', 'O'],  
 [ 'X', 'O', 'O', 'O', 'O'],  
 [ 'O', 'O', 'X', 'X', 'X'],  
 [ 'X', 'O', 'X', 'O', 'O']], 0, 0)  
Out [6]: False  
In [7]: har_nabo(  
[[ 'O', 'X', 'O', 'X', 'O'],  
 [ 'X', 'O', 'O', 'O', 'O'],  
 [ 'O', 'O', 'X', 'X', 'X'],  
 [ 'X', 'O', 'X', 'O', 'O']], 1, 3)  
Out [7]: True
```

- c) Skriv en funksjon `lag_kryssord(m, n)` som returnerer en todimensjonal liste svarende til et kryssordmønster med `m` rader og `n` kolonner. Om en rute er åpen eller lukket skal være tilfeldig, og sannsynligheten for at den er åpen skal være like stor som at den er lukket. Funksjonen skal returnere det genererte kryssordmønsteret. Eksempel på kjøring av funksjonen fra konsollen i

Spyder:

```
In [7]: kryssord = lag_kryssord(4, 5)
In [8]: kryssord
Out [8]:
[['0', 'X', '0', 'X', '0'],
 ['X', '0', '0', '0', '0'],
 ['0', '0', 'X', 'X', 'X'],
 ['X', '0', 'X', '0', '0']]
```

- d) Kryssordleverandøren vil ikke ha løsningsord på bare en bokstav. Derfor ber hun om at vi ikke lager kryssordmønster der det finnes åpne ruter uten åpne naboruter (over, under, til venstre eller til høyre). For et eksempel på en slik forbudt rute, se ruten i rad 0 og kolonne 0 (dvs. øverst i venstre hjørne) i kryssordet på slutten av oppgave 1c. Skriv en funksjon `fyll(kryssord)` som tar et rutemønster som parameter, og som lukker alle åpne ruter som ikke har åpne naboruter. Antall ruter som får ny verdi skal returneres. Eksempler på kjøring av funksjonen fra konsollen i Spyder (anta at `kryssord` først har verdien som vist på slutten av oppgave 1c):

```
In [9]: fyll(kryssord)
Out [9]: 1
In [10]: kryssord
Out [10]:
[['X', 'X', '0', 'X', '0'],
 ['X', '0', '0', '0', '0'],
 ['0', '0', 'X', 'X', 'X'],
 ['X', '0', 'X', '0', '0']]
In [11]: kryssord = [
    ['0', 'X', 'X', 'X'],
    ['X', '0', 'X', '0'],
    ['X', 'X', '0', 'X']]
In [12]: fyll(kryssord)
Out [12]: 4
In [13]: kryssord
Out [13]:
[['X', 'X', 'X', 'X'],
 ['X', 'X', 'X', 'X'],
 ['X', 'X', 'X', 'X']]
In [14]: fyll(kryssord)
Out [14]: 0
```