

Semesteroppgave 2

INF621 - Høstsemesteret 2021

Sammendrag

Semesteroppgavene er obligatoriske, og må være godkjente for å bestå INF621. Du skal levere én zip-fil, `sem2.zip`, som inneholder 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 9. desember kl 23:59

1 Grafiske temperaturreporter (50%)

Svar leveres på fil med navn `oppg1.py`

Den vedlagte filen `april.csv` inneholder maksimumstemperatur for fem norske byer nedlastet fra *Norsk Klimaservicesenter*. Hver linje inneholder feltene bynavn, dato og maksimumstemperatur den datoen.

1.a

(10%) Skriv en funksjon `les_temp(filnavn, skille=',')` med navnet på en CSV-fil som første parameter, og som leser filen. Andre parameter sier hvilket skilletegn som skal brukes. Funksjonen skal returnere en oppslagstabell der nøklene er bynavn, og verdiene er oppslagstabeller. I disse oppslagstabellene skal nøklene være dato, og verdiene maksimumstemperatur.

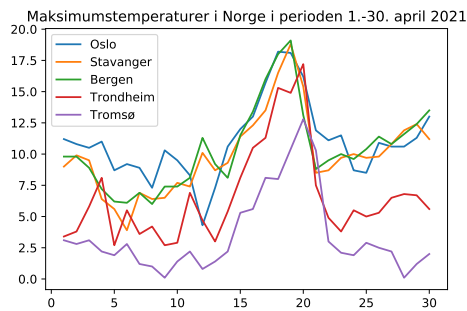
1.b

(20%) Skriv en funksjon `plott(makstemp)` som tar en oppslagstabell av typen forklart i forrige deloppgave som parameter. Funksjonen skal plote kurver av maksimumstemperatur for hver by som funksjon av tid. Alle kurvene skal framstilles i samme plott.

Kjøreeksempel

```
In [1]: makstemp = les_temp('april.csv', skille=',')
In [2]: plott(makstemp)
```

skal gi et plott av typen:



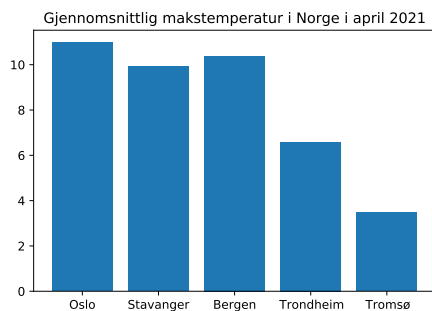
1.c

(10%) Skriv en funksjon `stolper_snitt(makstemp)` som lager et stolpediagram over gjennomsnittlig maksimumstemperatur i april for de fem byene. Funksjonen skal ta en oppslagstabell av typen nevnt i første deloppgave som parameter.

Kjøreeksempel

```
In [3]: stolper_snitt(makstemp)
```

skal gi diagram av typen:



1.d

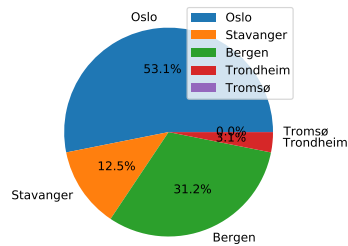
(10%) Skriv en funksjon `kake(makstemp)` som lager et sektordiagram over hvor ofte (hvor mange dager) hver by er den med høyest temperatur.

Kjøreeksempel

```
In [4]: kake(makstemp)
```

skal gi diagram av typen:

Hvor ofte byene har høyest temperatur i april 2021.



2 Sammenligning av nettaviser (50%)

Svar leveres på fil med navn `oppg2.py`

Vi vil analysere hvor ofte interessante ord forekommer på ulike avisers nettsider. For å forenkle analysen, vil vi

- ikke skille mellom store og små bokstaver,
- ikke tolke html-koden, men heller lete etter ordene i html-filen, og
- telle forekomster v.h.a. strengmetoden `count`, slik at f.eks. teksten '... kjenner lusa på gangen ...' gir en forekomst av søkeordet **USA**.

2.a

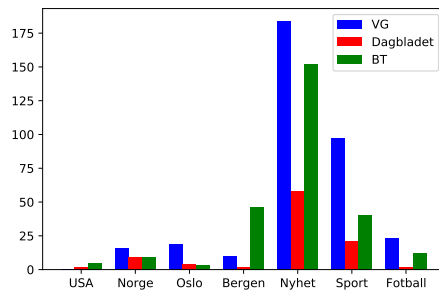
(20%) Skriv en funksjon `stolper(funn)` som tar som parametre en oppslagstabell (`dict`) over antall funn. Nøklene i `funn` er søkeord (strenger), og verdiene er oppslagstabeller med avisnavn som nøkler og antall forekomster av søkeordet i avisen som verdier.

Funksjonen skal lage et stolpediagram over antall funn av de ulike søkeordene i hver av avisene. Stolpene skal presenteres i grupper, med en gruppe for hvert søkeord. I hver gruppe skal det være en stolpe for hver avis, med ulike farger. Mellom hver gruppe skal det være et mellomrom like bredt som en stolpe.

Kjørekesempelet

```
In [1]: funn = {}
In [2]: funn['Norge'] = {'VG': 25, 'Dagbladet': 9, 'BT': 9}
In [3]: funn['Bergen'] = {'VG': 10, 'Dagbladet': 0, 'BT': 47}
In [4]: stolper(funn)
```

skal gi diagram av typen:



2.b

(20%) Skriv en funksjon `søk(søkeord)` som tar som parameter en liste med søkeord. Funksjonen går gjennom avisene VG, Dagbladet og BT, og lager en oppslagstabell, `funn`, over antall funn i hver av avisene av hvert søkeord i listen. (Utvalget av aviser kan for enkelhets skyld være hardkodet i funksjonen.) Deretter skal funksjonen kalle opp funksjonen `stolper` for å få grafisk framstilling av resultatet.

Kjørekesempelet

```
In [1]: søk(['Norge', 'Bergen'])
```

skal gi diagram av typen vist i forrige deloppgave.

2.c

(10%) Skriv kode for et `tkinter`-basert grafisk brukergrensesnitt (GUI) som inneholder en ramme med disse komponentene:

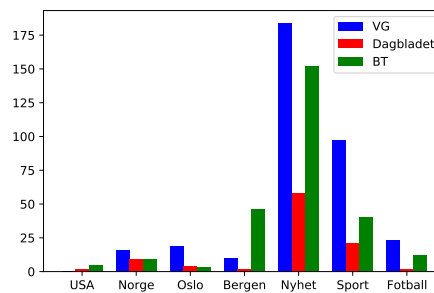
- en forklarende tekst (`Label`),
- en liste med mulige søkeord (strenger) brukeren kan velge fra (`Listbox`) og
- en knapp (`Button`).

Listen med mulige søkeord kan være hardkodet til

```
'Verden', 'USA', 'Europa', 'Norge', 'Noreg', 'Oslo', 'Bergen',
'Nyhet', 'Sport', 'Valg', 'Fotball', 'Klima', 'Corona', 'Korona'
```

Brukeren skal kunne velge et vilkårlig antall søkeord. Knappen skal ha en påskrift som f.eks. 'Start søk'. Den forklarende teksten skal informere brukeren kort om at hun kan velge interessante søkeord, og deretter trykke på knappen for å lansere søket. Knappen skal v.h.a. parameteren `command` kobles til en funksjon `start_søk` som skal lage en liste av utvalgte søkeord, og sende listen til funksjonen `søk` fra forrige deloppgave. Resultatet skal dermed bli et stolpediagram over funn av utvalgte søkeord i de tre avisene. Merk at du også må skrive funksjonen `start_søk`.

Eksempel: Dersom brukeren velger søkeordene som vist i figuren til venstre, og deretter trykker på 'Vis resultat', skal et stolpediagram av typen vist i figuren til høyre dukke opp:



Hint: Klassen `Listbox` har en metode `curselection()` som returnerer et tuppel bestående av indeksene til de strengene som bruker har valgt.