

Alle oppgavene skal demonstreres til en studentassistent på sal. I oppgaver der du skriver programkode skal også denne vises fram. Lykke til!

1 Generelt om lister

En tom liste i Python kan lages ved å skrive `[]`. Vilkarlig lange lister kan lages ved å skrive `[a, b, c, ...]`, der `a`, `b`, og `c` er elementer i listen.

Eksempel 1

For å lage en liste `l` med elementene 3, 5, og -41, skriver du `l = [3, 5, -41]`.

Elementene i en liste kan leses ved å skrive nummeret til elementet i firkantklammer etter navnet på listen. Dette kalles å indeksere listen. Lister kan også indekseres baklengs fra slutten av listen. Dette gjøres ved å putte et negativt tall inni firkantklammene.

Kodesnutt 1

```
liste = [1, 2, 3, 4, 5]

en = liste[0]
to = liste[1]
fem = liste[-1]
```

- a) Lag listen `li` med alle heltallene fra 1 til og med 6.
- b) Gang alle partallene i samme liste med `-1` og lagre de i samme posisjon i listen.
- c) Sorter listen omvendt (dvs. høyest til lavest) ved hjelp av pythonlisters innebygde sorteringsmetode.
- d) Skriv listen ut til skjermen.

2 Lister som vektorer

En liste kan brukes til å representere en matematisk vektor.

En av forskjellene mellom lister og vektorer er at lister kan inneholde elementer av forskjellige typer (f.eks både tall og strenger). I denne oppgaven skal du bruke lister til å implementere tredimensjonale vektorer, altså vektorer med tre elementer.

En vektor $\mathbf{x} = (x, y, z)$ har lengde $\sqrt{x^2 + y^2 + z^2}$. Den kan multipliseres med tall, og vektor-multipliseres med andre vektorer.

Multiplikasjon med tall, også kalt skalarmultiplikasjon, ganger hver komponent i vektoren med tallet.

Multiplikasjon mellom vektorer \mathbf{u} og \mathbf{v} av tre dimensjoner er definert som:

$$\mathbf{u} \cdot \mathbf{v} = u_1v_1 + v_2u_2 + v_3u_3$$

Eksempel 2

$$[3, 7, 1] \cdot [2, -4, 8] = 3 \cdot 2 + 7 \cdot (-4) + 1 \cdot 8 = -14$$

Multiplikasjon av vektorer kalles også for prikkprodukt (dottprodukt), skalarprodukt, eller indreprodukt.

- Lag en funksjon program som tar inn de tre komponentene til en tre-dimensjonal vektor (**på én linje**), og returnerer de som en liste.
- Lag en funksjon som skriver ut vektoren på en vakker måte. Les inn en vektor `vec1` med funksjonen i punkt a) og skriv vektoren til skjermen.
- Lag en funksjon som tar inn en vektor og en skalar som argumenter, skalarmultipliserer vektoren og returnerer den nye vektoren.
Les inn en skalar fra brukeren og bruk funksjonen til å skalarmultiplisere `vec1`. Tilegn resultatet til `vec1`. Skriv ut den nye `vec1`.
- Lag en funksjon som tar inn en vektor som argument og regner ut dens lengde. Utvid programmet til å regne og skrive ut vektorlengden både før og etter skaleringen i forrige deloppgave. Skriv også ut forholdet mellom de to lengdene.
- Lag en funksjon som tar inn to vektorer som argumenter og returnerer indreproduktet av de to vektorene.
Utvid programmet til å lese inn enda en vektor `vec2`. Kall funksjonen med de to vektorene og skriv ut resultatet.

3 The passion of the tooth fairy

Overformynderiet for tannfeer og Java-programmerere har vedtatt at isteden for å betale 10 kroner for hver enkelt tann et barn mister og putter i vannglasset sitt, er de nå nødt til å betale per gram tann. Det ble bestemt at ett gram tann er verd 50 øre¹.

Overformynderiet bestemte også at 50-øringen skal gjeninnføres som gyldig betalingsmiddel.

50-øringer består stort sett av kobber. Dessverre er alle tannfeer allergisk mot kobber, og vil helst bruke større mynter når de kan. Du må derfor fordele myntene i hvert vannglass fra større verdi til minst.

¹Tannstøv har steget ganske kraftig på børsen i det siste.

Gitt følgende liste der hvert tall representerer en lite barns tannvekt i gram, hvor mange 20, 10, 5, 1, og 0.5-mynter må tannfeen bringe med seg for å dele ut for **hvert** barn?

Kodesnutt 2

```
teeth = [  
    95, 103, 71, 99, 114, 64, 95, 53, 97, 114,  
    109, 11, 2, 21, 45, 2, 26, 81, 54, 14,  
    118, 108, 117, 27, 115, 43, 70, 58, 107]
```

4 Mini-Eliza

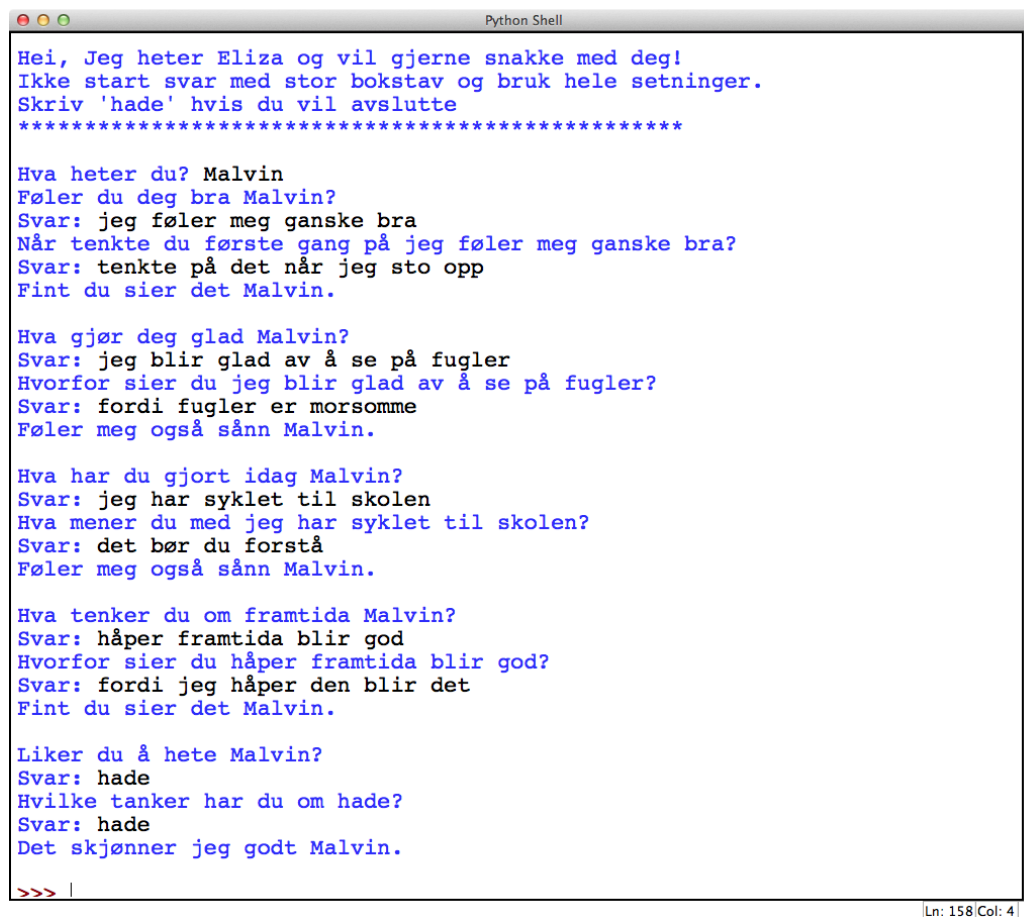
I denne oppgaven skal du lage et program som simulerer en datavenn du kan snakke med. En del av koden finnes allerede og din oppgave er å skrive koden som mangler for å fullføre programmet.

Hovedfunksjonen til programmet er følgende:

1. Spørre brukeren om navnet.
2. Stille et tilfeldig spørsmål fra ei liste av spørsmål til brukeren der navnet inngår.
3. La brukeren få svare.
4. Stille et tilfeldig oppfølgingsspørsmål, der svaret fra brukeren er med.
5. Gi en tilfeldig kommentar til brukeren.

Programmet skal avsluttes ved at brukeren gir svaret “hade”. Kildekoden er godt dokumentert med kommentarer og alle steder som mangler kode er markert med “MANGLER KODE HER!!!”.

En kjøring av et fungerende program kan se ut som i figur 1.



```
Python Shell

Hei, Jeg heter Eliza og vil gjerne snakke med deg!
Ikke start svar med stor bokstav og bruk hele setninger.
Skriv 'hade' hvis du vil avslutte
*****

Hva heter du? Malvin
Føler du deg bra Malvin?
Svar: jeg føler meg ganske bra
Når tenkte du første gang på jeg føler meg ganske bra?
Svar: tenkte på det når jeg sto opp
Fint du sier det Malvin.

Hva gjør deg glad Malvin?
Svar: jeg blir glad av å se på fugler
Hvorfor sier du jeg blir glad av å se på fugler?
Svar: fordi fugler er morsomme
Føler meg også sånn Malvin.

Hva har du gjort idag Malvin?
Svar: jeg har syklet til skolen
Hva mener du med jeg har syklet til skolen?
Svar: det bør du forstå
Føler meg også sånn Malvin.

Hva tenker du om framtida Malvin?
Svar: håper framtida blir god
Hvorfor sier du håper framtida blir god?
Svar: fordi jeg håper den blir det
Fint du sier det Malvin.

Liker du å hete Malvin?
Svar: hade
Hvilke tanker har du om hade?
Svar: hade
Det skjønner jeg godt Malvin.

>>> |
```

Ln: 158|Col: 4

Figur 1: Kjøring av kode