

Bonusoppgaver

I denne oppgavesamlingen finner du et knippe med bonusoppgaver for dag 1 dersom du vil ha litt mer å bryne deg på. God koding!

Oppgave 1 *Jordkloden*



I denne oppgaven skal vi øve på å bruke Python som kalkulator, ved å regne litt på jordkloden. Husk at formelen for volumet av en kule er

$$V = \frac{4}{3}\pi R^3.$$

- a) Jordkloden er tilnærmet en perfekt kule, og har en radius på 6371 km. Lag et kort program som først definerer en variabel `radius`, og deretter regner ut en variabel `volum`. Skriv til slutt ut svaret til brukeren med `print()`-funksjonen. La svaret være i km^3 .
- b) Endre programmet ditt så svaret istedet skrives ut i antall liter.
- c) Den totale massen til jordkloden er omtrent $M = 5.972 \cdot 10^{24}$ kg. Regn ut hvor mange kg hver liter av jordkloden veier i gjennomsnitt. Virker svaret ditt rimelig?

Oppgave 2 *Poengtelling med while og input*

Oskar og Li Min spiller et spill hvor man i løpet av spillet vinner briketter for *plusspoeng* og *minuspoeng* som man teller opp ved slutten av spillet. For å holde orden på poengene når de teller vil Li Min lage et lite Python program.

- a) Opprett en variabel `total_poengsum` som starter på 0
- b) Bruk `input` til å spørre en bruker om de ønsker å legge til et plusspoeng, trekke fra et minuspoeng eller avslutte
- c) Bruk en betingelse slik at `total_poengsum` øker med 1 dersom brukeren skrev inn `'pluss'` og synker med 1 dersom brukeren skrev `'minus'`
- d) Bruk en `else`-blokk til å gi beskjed dersom brukeren skrev inn noe ugyldig
- e) Bruk en `while`-løkke slik at programmet fortsetter å spørre helt til brukeren svarer `'avslutt'`. Når løkka er ferdig skal den totale poengsummen skrives ut. (**Hint:** for å unngå uendelig løkke må du passe på å spørre med `input` inne i løkka!).

Oppgave 3 `and` og `or`

I denne oppgaven skal vi se på hvordan `and`- og `or`-operatorene fungerer og hvordan de brukes i betingelser.

- a) Lag en variabel som inneholder tallet 5. Bruk en betingelse til å teste om tallet er mindre enn 10, og print isåfall ut en melding.
- b) Bruk nøkkelordet `and` til å teste om tallet er mindre enn 10 og større enn 2.
- c) Bruk nøkkelordet `or` til å teste om tallet er mindre enn 6 eller større enn 10.

Oppgave 4 *Brette ark*

Et vanlig ark er omtrent 0,1 mm tykt. Om vi bretter arket på midten dobblør vi tykkelsen av arket, så det er 0,2 mm tykt. Om vi bretter arket på nytt dobblør vi igjen tykkelsen, så det blir 0,4 mm tykt. Sånn kan vi fortsette å brette arket for å gjøre det tykkere. Om du prøver i praksis viser det seg nok fort at det er veldig vanskelig å brette arket noe særlig mer enn 6-7 ganger. Men om vi nå later som vi kunne brettet arket så mange ganger vi vil, er det ingen grense for hvor tykt arket kunne blitt.

Verdens høyeste bygning er Burj Khalifa i Dubai, som er 828 meter høyt. Vi

ønsker å finne ut hvor mange ganger vi må brette arket vårt, før det er like tykt som høyden av denne bygningen.

- a) Diskuter med sidemannen hvordan dere kunne funnet ut av dette med penn og papir. Hva er ulempen med fremgangsmåten deres?

Vi skal nå løse problemet ved hjelp av et kort Python-program. For å løse problemet bruker vi en løkke for å brette arket helt til vi har nådd den tykkelsen vi er ute etter.

- b) Fyll inn skjelettkoden under for å finne antall brett vi trenger. Pass spesielt på at tykkelsen til arket er oppgitt i millimeter, mens bygningen er oppgitt i meter.

```
1 tykkelse = ...
2 antall_brett = ...
3
4 while ...:
5     tykkelse *= ...
6     antall_brett += ...
7
8 print(...)
```

- c) Om du har klart å løse oppgaven. Finn antall brett som skal til før tykkelsen er like høyt som verdens høyeste fjell, Mount Everest, som er 8848 meter høyt.
- d) Avstanden fra jorda til månen er ca 384400 km. Hvor mange ganger må arket brettes før det er like tykt som denne avstanden?

Oppgave 5 *Utfordring: Chatbot*

Mange selskaper har en chatbot som kunder kan stille spørsmål til, før de eventuelt blir sendt videre til mennesker som kan hjelpe dem over nett eller telefon. Dette er en fin måte å spare ressurser på, da en chatbot kan programmeres til å svare på de vanligste spørsmålene, slik at bare mer avanserte

problemer trenger å behandles av et kundesenter.

Noen chatboter er mer avanserte enn andre, men med hjelp av kun betingelser, kan man lage en helt fungerende chatbot som kan svare på et sett med spørsmål.

I denne oppgaven kan du enten finne på ditt eget problem, eller ta utgangspunkt i det som presenteres på neste side. Eksempler man kan prøve seg på er: en guide til å finne ting på et bibliotek, en chatbot for å bestille mat på en restaurant, eller en chatbot som hjelper deg å leie en bil.

I dette eksempelet foreslår vi å lage en chatbot for nettsiden til en bank, som kan svare på de vanligste spørsmålene kunder har.

Uansett om du bruker dette eksempelet eller ditt eget, er det lurt med samme fremgangsmetode. Tegn først opp et tre eller en liste, som beskriver utfallene, og hva de skal føre til. Skal de føre til svar, eller nye spørsmål? Det kan være lurt å nummerere utfall, slik at brukeren bare svarer 1,2 eller 3 osv.

- a) Tenk deg tre alternative spørsmål eller temaer, som brukeren kan få svar på, og noter ned hva chatbotten skal gjøre i de ulike tilfellene. Du kan ta utgangspunkt i eksempelet med en bank-bot på neste side.
- b) Sett opp den første *if*, *elif*, *else* - betingelsen som lar brukeren velge mellom de tre hovedalternativene. På neste side ser du en eksempelkode på hvordan dette kan gjøres.
- c) Utvid programmet programmet ditt, slik at de ulike emnene forgreiner seg til flere spørsmål og svar, ved å bruke betingelser inni betingelser.
- d) La noen prøve chatboten din!

```
<Hyggelig beskjed>
<Hva kan jeg hjelpe deg med>
|- 1.LÅN OG SPARING
   |- 1. Renter          -->    <Vi kan tilby disse
                                   ulike lånene med
                                   disse rentene,...

   |- 2. Opprette        -->    <Gå til denne siden
                                   for å opprette en
                                   sparekonto hos oss>
```

```
| - 2. NETTBANK
    | - 1. BankID      -->  <Kontakt vårt kundesenter
                             på chat eller tlf
                             mellom kl.8-16 på hverdager

    | - 2. Konto        -->  <For hastesaker, ring
                             døgnåpent nummer (nummer).
                             Ellers, kontakt vårt
                             kundesenter

| - 3. SPERRING AV KORT  -->  <Ring oss på (nummer)
                             så fort som mulig
```

Eksempelkode:

```
1 print('<hyggelig beskjed>')
2 svar = input(''Hva kan jeg hjelpe deg med?
3         1. <Alternativ 1>
4         2. <Alternativ 2>
5         ... ''')
6 if svar == '1':
7     <svar, eller still nytt spørsmål>
8 elif svar == '2':
9     <svar, eller still nytt spørsmål>
10 elif svar == ... :
11
12 else:
13     <alternativ beskjed>
```