

# Oppgaver for dag 1

Her er et knippe oppgaver for dag 1 av kodekurset. Tema for første dag er variabler, input, while-løkker og betingelser. Dersom du står fast er det bare å spørre, kursholderene er den viktigste ressursen dere har, benytt oss mens vi er her!

God koding!

## Introduksjon til Python

### Oppgave 1 *Printing*

- a) Lag et program som skriver ut teksten «Hei, verden!» til skjermen.
- b) Lag et program der du først lagrer navnet ditt i en variabel, og så få programmet ditt til å skrive ut en hilsen direkte til deg.

### Oppgave 2 *Finn fire feil!*

Her følger det fire programmer som har blitt skrevet feil. Finn feilen i hver programsnutt. Du kan godt kjøre programmet inn på din egen maskin, og kjøre det, da kan kanskje feilmeldingen hjelpe deg å skjønne hva som er galt.

Når du tror du skjønner hva som er galt, rett feilen på din egen maskin, og kjør programmet for å sjekke at det fungerer som det skal.

a)

```
1 Print('Python er gøy!')
```

b)

```
1 navn = "Margaret"
2 print(navn)
```

c)

```
1 frukt = 'eple'
2 print(eple)
```

d)

```
1 print(Hei på deg)
```

### Oppgave 3 *Flere variabler*

Følgende program skriver ut det den vet om en person:

```

1 navn = "Amalie"
2 favorittfarge = "gul"
3 favorittall = 4
4 favorittland = "Spania"
5
6 print(f"Hei, {navn}! Her er det jeg vet om deg:")
7 print(f"* Din favorittfarge er {favorittfarge}.")
8 print(f"* Ditt favorittall er {favorittall}.")
9 print(f"* Ditt favorittland er {favorittland}.")

```

Kopier eller skriv av programmet, og sjekk at det kjører.

- a) Endre på variablene slik at programmet skriver ut ditt navn, din favorittfarge, ditt favorittall og ditt favorittland.
- b) Hvis du har denne oppgaven i utskriftsversjon – lag piler på arket ditt som peker *fra* linjen der hver variabel deklarerer og *til* den linjen den påvirker utskriften (der den er med i en `print`).
- c) Prøv å endre på rekkefølgen i programmet ved å flytte rundt på linjene, og se hva som skjer. Hvilke linjer *må* komme før en annen linje, og hvilke linjer *kan* komme i vilkårlig rekkefølge? Prøv å forklare sammenhengene du ser til deg selv eller en annen person.

#### Oppgave 4 *Regne total lønn*

Koden under ber en bruker om en timelønn og antall timer og regner ut total lønn

```
1 timelønn = int(input("Hva er timelønna di?"))
2 antall_timer = int(input("Hvor mange timer har du
   jobbet?"))
3
4 total_lønn = timelønn*antall_timer
5
6 print(f"Den totale lønna er {total_lønn}")
```

- a) Kjør programmet og prøv ut forskjellige timelønner og timeantall
- b) Endre programmet slik at det isteden tar inn timelønn, antall arbeidsdager og antall timer per arbeidsdag og regner ut total lønn basert på det

## Variabler og regning

### Oppgave 5 *Arealberegninger*

Formlene for å regne ut arealet av et rektangel er gitt ved følgende formel:

$$A = lh, \quad (1)$$

der  $l$  og  $h$  står for lengde og høyde, respektivt.

- a) Skriv et program som inneholder variablene lengde og høyde, og gi de verdier som du velger selv, si i enheten cm.
- b) Opprett en variabel `areal_rektangel`, definert ved hjelp av lengde og høyde, dvs. skriv inn

```
1   areal_rektangel = lengde*høyde
```

- c) Skriv ut arealet av rektangelet til terminalen på en hensiktsmessig måte.
- d) Omkretsen til et rektangel er gitt ved  $O = 2l + 2h$ , dvs. 2 ganger lengde og 2 ganger høyde. Definer en variabel `omkrets_rektangel` på samme måte som du definerte `areal_rektangel`, og skriv denne ut i tillegg.
- e) Prøv å endre på lengde og høyde og se hva du får av utskrift for forskjellige verdier.

### Oppgave 6 *Regne mellom SI-enheter*

En centimeter er 0.01 meter. En millimeter er 0.1 centimeter. En mikrometer er 0.001 millimeter.

- a) Lag en variabel med din høyde i meter (m), og lag en ny variabel som gjør denne høyden til cm.
- b) Lag en ny variabel som regner om høyden din i cm til høyden din i mm.
- c) Lag enda en variabel som regner om høyden din i mm til høyden din i  $\mu\text{m}$ .

- d) For hver av enhetene over, skriv ut resultatet med en hensiktsmessig utskrift.

### Oppgave 7 *Konvertering av temperatur*

I Norge oppgir vi temperaturer i målestokken *celsius*, men i USA bruker de ofte målestokken *fahrenheit*. Hvis du finner en kakeoppskrift fra USA kan det for eksempel stå at du skal bake kaken ved 350 grader. Da mener de altså 350°F. Vi vil nå lage et verktøy som kan konvertere denne temperaturen for oss, sånn at vi vet hva vi skal bake kaken ved i celsius..

For å regne over fra fahrenheit til celsius bruker vi formelen:

$$C = \frac{5}{9}(F - 32).$$

Der  $F$  er antall grader i Fahrenheit, og  $C$  blir antall grader i celsius.

- a) Start med å opprette en variabel, *fahrenheit*, som du setter lik 350.
- b) Lag et program regner ut hvor mange grader celsius 350 °F tilsvarer. Virker det rimelig å skulle bake en kake ved denne temperaturen?

Programmet du har lagd tar en temperatur i fahrenheit, og gjør om til celsius. Men hva om vi ønsker å gå motsatt vei? Om vi ønsker å lage et nytt program som gjør motsatt, så må vi først ha en formel for  $F$ .

- c) Klarer du å ta uttrykket

$$C = \frac{5}{9}(F - 32).$$

og løse for  $F$ ?

- d) Lag et nytt program (som du lagrer som en egen fil) som har en variabel, *celcius*, som du setter lik 220. Regn så ut hvor mange grader fahrenheit dette er, og skriv det ut til brukeren.
- e) Modifiser programmet ditt til å finne frysepunktet og kokepunktet til vann i målt i fahrenheit.

## Mer om Input og konvertering

### Oppgave 8 *Kulegeometri*

Under er en kodesnutt:

```
1 from pylab import pi
2 radius = float(input("Hva er radiusen? "))
3 kulevolum = (4/3)*pi*radius**3
4 print(kulevolum)
```

Kopier over eller skriv av koden til et Python-program, og prøv å kjøre det.

- Hva gjør koden? Forklar hva som skjer, linje for linje.
- Modifiser koden til å ha en mer beskrivende utskrift.
- Modifiser koden så den regner ut og skriver ut overflateareal i tillegg til volum.
- Lag et nytt Python-program som ber en bruker om å skrive inn radius og høyde til en sylinder og skriver ut tilhørende volum og overflateareal.

### Oppgave 9 *Finn tre feil!*

Her følger det tre programmer som ikke fungerer. Skriv programmet inn på din egen maskin og kjør det, les feilmeldingen, og prøv å tolke den. Når du skjønner hva som er galt, rett opp feilen og kjører programmet.

a)

```
1 ukedag = input(Hvilken dag er det i dag?)
2 print(ukedag)
```

b)

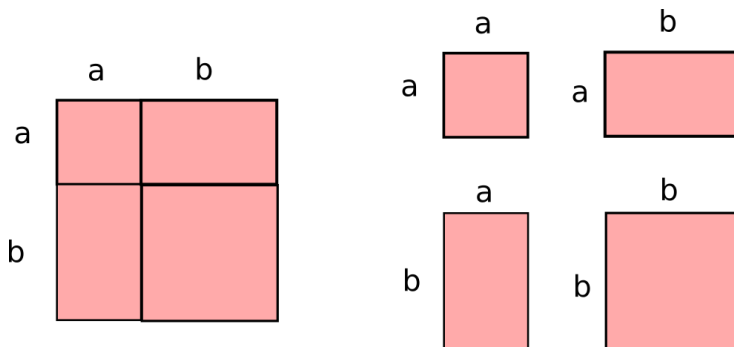
```
1 navn = input("Hva heter du?")
2 print(name)
```

c)

```
1 input("Hvor gammel er du?")
2 print(alder)
```

### Oppgave 10 *ab-kvadrat*

Anta at vi har et kvadrat med sidelengder  $a + b$ . Vi kan dele opp kvadratet i fire deler, på denne måten:



– dvs. i fire deler, der hver del har totalt areal bestemt av  $a$  og  $b$  som vist til høyre.

- Lag et program som tar inn to tall og lagrer de i variabler  $a$  og  $b$ . Gjør om tallene til desimaltall ved hjelp av `float`.
- Regn ut arealet av delen med areal  $a \cdot a$ , lagre svaret i en variabel og skriv den ut. Regn tilsvarende ut arealet av de andre delene, og skriv ut disse på samme måte. Lagre hvert svar i egen variabel (du kan bruke samme utregning og variabel for å regne ut  $a \cdot b$  som for  $b \cdot a$ ).
- Regn tilsammen ut det totale arealet av alle delene ved å plusse de ulike delene sammen. Lagre svaret i en variabel. Skriv ut totalarealet.
- Totalarealet du fikk i forrige oppgave burde bli det samme som  $(a+b)^2$ , altså

$$(a+b) \cdot (a+b)$$

Regn ut totalarealet på denne måten også. Dobbeltsjekk svaret ditt fra forrige oppgave ved å sjekke at du får samme svar.



## Betingelser

### Oppgave 11 *Sjekk alder*

En film på kino har aldersgrense 15 år. Vi skal lage et program som interagerer med brukeren, og finner ut om de kan se filmen eller ikke.

- a) Lag et program som printer ut en hilsen til brukeren, og spør hvor gamle de er. Lagre alderen i variabelen `alder`.
- b) Lag en test som sjekker om brukeren er 15 år eller eldre. Skriv ut passende svar avhengig av om de er gamle nok til å se filmen eller ikke.
- c) Utvid programmet ditt til å regne ut hvor mange år det er til brukeren kan se filmen.
- d) La programmet ditt gjøre et unntak for de som er 12 år eller eldre dersom de har med seg en voksen.

### Oppgave 12 *Vinkeltyper*

Vi kan dele vinkler inn i tre forskjellige typer:

1. En vinkel som er mindre enn  $90^\circ$  kalles en *spiss* vinkel.
2. En vinkel som er større enn  $90^\circ$  kalles en *stump* eller *butt* vinkel.
3. En vinkel som er akkurat  $90^\circ$  kalles en *rett* vinkel

Be brukeren om å gi en vinkel og bruk `if`, `elif` og `else` til å fortelle brukeren om vinkelen er *spiss*, *stump* eller *rett*

### Oppgave 13 *Betingelser for kjemiske tilstander*

Programmet under ber en bruker om temperaturen til vann og skriver ut om det er flytende, fast eller gass basert på temperaturen

```

1 kokepunkt = 100 # grader celcius
2 frysepunkt = 0 # grader celcius
3 stoff_navn = 'vann'
4
5 temperatur = float(input(f'Hvilken temperatur har {
    stoff_navn}et?'))
6
7 if temperatur < frysepunkt:
8     print('Det er et fast stoff')
9 elif temperatur < kokepunkt:
10    print('Det er en væske')
11 else:
12    print('Det er en gass')

```

- Hva brukes variabelene frysepunkt, kokepunkt og stoff\_navn til?
- Utvid programmet slik at det også skriver ut hvor mange grader under kokepunktet og frysepunktet vannet er.
- Kvikksølv har frysepunkt -39 grader celcius og kokepunkt 357 grader celcius. Modifiser programmet slik at det gjelder for kvikksølv istedenfor vann.

## while-løkker

### Oppgave 14 Enkle while-løkker

Bruk en **while**-løkke til å løse følgende oppgaver:

- Print ut meldingen 'Hei, verden' 6 ganger.
- Print ut alle tallene fra 0 til 100.
- Print ut alle oddetallene fra 0 til 101.

### Oppgave 15 Gangetabellen

Lag et program som ber brukeren om å sende inn et tall fra 1-10, og deretter printer ut én rad i gangetabellen, tilsvarende tallet som ble skrevet inn, ved hjelp av en **while**-løkke. Lag et nytt program som printer ut hele den lille gangetabellen ved hjelp av to **while**-løkker.

**a) Oppgave 16** *Brette ark*

Et vanlig ark er omtrent 0,1 mm tykt. Om vi bretter arket på midten dobblar vi tykkelsen av arket, så det er 0,2 mm tykt. Om vi bretter arket på nytt dobblar 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 17 *Vekst i folketall*

Folketallet i<sup>a</sup> og den prosentvise veksten<sup>b</sup> i folketall i utvalgte land i 2020 var som følger:

Land	Folketall	Vekst
Australia	25 687 040	1.3 %
Brasil	212 559 410	0.7 %
Kenya	53 771 300	2.3 %

I denne oppgaven skal vi regne litt på forventet vekst ved hjelp av disse tallene.

- a) Lag et Python-program der du oppretter variabler for folketallet i hvert av landene, og variabler for den prosentvise veksten.
- b) Anta at befolkningen i hvert land får samme vekst i 2021. Utvid programmet ditt slik at det regner ut og skriver ut hva befolkningen i hvert land vil være i 2021.
- c) Anta videre at det samme skjer for hvert år fem år frem i tid. Lag en `while`-løkke som regner ut og skriver ut den oppdaterte befolkningen, der vi hele tiden antar at befolkningsveksten er konstant.

- d) Vi snur problemet på hodet, og spør oss selv: Når vil et visst land nå et visst folketall, igjen antatt konstant prosentvis vekst? Lag et nytt program, og ta vare på variablene for Australia. La programmet regne ut og skrive ut hvor mange år det tar til Australia vil få over 26 millioner innbyggere.
- e) Modifiser programmet ditt til å finne ut hvor mange år det tar før Australia får over 30 millioner innbyggere.
- f) Modifiser programmet ditt igjen (eller kopier over til et nytt program for Brasil) slik at du kan finne ut hvor mange år det tar til Brasil når 213 millioner innbyggere, og 220 millioner innbyggere.
- g) Modifiser programmet ditt igjen slik at du kan finne ut hvor mange år det tar til Kenya når 54 millioner innbyggere, og 60 millioner innbyggere.

---

<sup>a</sup>Kilde: <https://data.worldbank.org/indicator/SP.POP.TOTL>

<sup>b</sup>Kilde: <https://data.worldbank.org/indicator/SP.POP.GROW>