

IN1000 – SEMINAR 5

- Repetisjon: Funksjoner og prosedyrer
 - Skop
 - Fil

Funksjon	<p>En funksjon som defineres med <i>def</i>, som ikke er del av en klasse (ordet <i>self</i> brukes ikke). Funksjoner har alltid en returverdi.</p> <p>Eks:</p> <pre>def sum(a, b): c = a + b return c</pre> <p>På engelsk kalles dette <i>function</i>.</p>
Prosedyre	<p>Tilsvarende funksjon, men uten returverdi. Bruker aldri ordet <i>self</i>, og er ikke del av en klasse.</p> <p>Eks:</p> <pre>def superprint(ord): print("ordet er", ord)</pre> <p>På engelsk kalles dette <i>procedure</i>.</p>
Metode	<p>Tilsvarende funksjon, men som del av en klasse. Har alltid <i>self</i> som første parameter. Kan, men må ikke, ha en returverdi.</p> <p>Eks:</p> <pre>def areal(self): firkant_areal = self.lengde * self.bredde return firkant_areal</pre> <p>På engelsk kalles dette <i>method</i>.</p>

Funksjoner repetisjon

- Kan ha så mange parametere den vil ha, eller ingen.
- Returnere kun en ting
 - *Hvis man vil returnerer mer enn en ting må man lage en liste, legge tingene i listen og returnere hele listen*
 - *Som regel når man kaller en funksjon tar man vare på verdien den returnerer med en variabel*

Oppgave 1

Skriv en funksjon som tar i mot to tall og returnerer det største av de to tallene (ikke skriver den ut). Lag en main prosedyre som kaller på funksjonen din og skriver ut resultatet av å bruke funksjonen.

Skop

- Den delen av programmer som kan aksessere/få tilgang på en variabel
- Lokal variabel: en variabel som er definert inne i en prosedyrer/funksjon.
 - *Kun denne prosedyren/funksjon har tilgang på variabelen*
- Global variabel: defineres utenfor en funksjon/prosedyre
 - *Alle prosedyrer/funksjoner har tilgang på variabelen*

- PS: Generelt burde man bruke lokale variabler og heller lage prosedyrer som returnerer verdier fremfor globale variabler som oppdateres av mange prosedyrer.

```
def summer():  
    print("sum: ", a + b)
```

```
a = 3  
b = 2  
summer()
```

```
def minus():
```

```
    a = 3  
    b = 2
```

```
def summer():
```

```
    print("sum: ", a + b)
```

```
summer()
```

Filer og tekster

- Lese fra fil:
 - `innfil = open("filnavn.filtype", "r")` # r står for read
 - `linje = innfil.readline()` #leser en linje fra en oppgitt fil. `Readline()` returner en string
- Skrive til fil:
 - `utfil = open("filnavn.filtype", "w")` #w står for write
- Legge til noe i en fil:
 - `utfile = open("filnavn.filtype", "a")` "a står for append
- Open returnerer et filobjekt som har metoder vi kan bruke for å lese eller skrive til filen
- Lukke fil:
 - `innfil.close()`
 - `utfil.close()`
 - `close()` lukker filen

Ulike måter å lese fra fil

- `readline()`: leser en linje fram til `\n`(newline)
 - `linje = minFil.readline()`
 - *returnerer en string*
- `readlines()`: leser hele filen
 - `listeMedLinjer = minFil.readlines()`
 - *returnerer en liste hvor hvert element er en linje i filen (i rekkefølge)*
- gå gjennom filen med en for-løkke:
 - *for linje in minFil:*

Eksempel: readline()

```
minFil = open("universiteter.txt", "r")

linje = minFil.readline()
while linje != "":
    print("Universitet:", linje)
    linje = minFil.readline()

minFil.close()
```

Eksempel: or linje in <filnavn>

```
minFil = open("universiteter.txt")

for linje in minFil:
    print("Universitet:", linje)

minFil.close()
```

Eksempel: skrive til fil

```
minFil = open("universiteter.txt", "w")

minFil.write("Universitetet i Oslo(UiO)\n")
minFil.write("oslomet\n")

minFil.close()

minFil = open("universiteter.txt", "a")

minFil.write("Universitetet i Bergen(UiB)\n")
minFil.write("Universitetet i Stavanger(UiS)\n")

minFil.close()
```

Oppgave 2

Anta at du har filen “historie.txt”. Les inn alle linjene og lagre dem i en liste. Første linje skal ligge først i listen, osv. (Tips: husk å åpne og lukke filen..)

Oppgave 3

Skriv en funksjon som tar i mot to sannhetsverdier (True/False) og returnerer 1 hvis begge verdier er sanne, 0 hvis begge verdier ikke er sanne.

Gruble:

Hva tar ikke funksjonen høyde for? Hva kan gå "galt"?

Oppgave 4

Skriv et program som bruker for-løkker til å ...

- a) Skriv ut alle partall mellom 0 og 10.
- b) Finne summen til alle partall mellom 0 og 10.
- c) Finne summen av alle oddetall mellom 10 og 0

Oppgave 5

- a) Skriv et program som definerer en liste [2, 3, 6, 8]. Bruk en for-løkke til å skrive ut alle verdiene i listen.
- b) Skriv om prosedyren over/skriv ny prosedyre, som tar en liste som input/parameter og som skriver ut alle verdiene i listen.
- c) Endre prosedyren til en funksjon slik at den returnerer den laveste verdien i listen i stedet for å skrive dem ut.