

Velkommen :)

Seminartime - IN1000 Gruppe 15 Digøk

srhellst på Matternmost
Epost: srhellst@ifi.uio.no



UiO : Universitetet i Oslo

Oppgave 1

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

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

Se på følgende kodelinjer, beskriv kodeflyten linje for linje, nevnt de relevante globale og lokale variablene, og forklar hva som skrives ut.

```
a = 9
b = 5

def foo(a):
    if a > 5:
        bar()
    else:
        baz(a)

def bar():
    if a >= 8:
        print(a)
    else:
        print("***")

    return 5

def baz(b):
    c = b + bar()
    print(c)

a = 8

foo(7)
foo(4)
```

Oppgave 5 (EKSTRA OPPGAVE, utfordrende)

Denne oppgaven er åpen, dere får et problem og skal velge litt hvordan dere vil løse det selv (les hele oppgaven før du startet).

Oppgave: Grunnet koronna har man i flere gruppetimer ha oppmøtere registrering. Gitt filen gruppeXNavn.txt spør brukerne (i terminalen) om brukeren er til stedet, hvis brukeren taster inn x er studenten der, ellers er studenten ikke stede. Oppmøtet skal registreres i en annen fil som heter uke<ukenummer>.txt (du må spørre brukeren om hvilken uke det er). Her skal alle navnene skrives til fil, men hvis brukeren er til stede skal det stå en x bak navnet. Hvilket nummer studenten er skal stå foran navnet.

Filen som man skriver ut skal f.eks. Se slik ut:

1. Tiril Torgren: x
2. Sandra Frogn:
3. Emil Orgensbø: x

Oppgave 6

I denne oppgaven skal dere lage et program som leser en fil med studenters navn og deres karakterer.

Linjene i filen ser slik ut:

```
Anna,C  
Joakim,E
```

a)

Lag en funksjon `sorter_etter_karakter()` som tar inn parameter filnavn og leser hver linje i filen i en for-løkke.

- Inne i for-løkken, hent navn og karakter fra linjen (uten linjeskrift-karakter).
- Lagre navnet og karakteren i et ordbok som bruker karakteren som nøkkelverdi, og har en liste med navner som innholdsverdi.
- Hver gang du finner et karakter som ikke er en nøkkel i ordboen ennå, kan du legge det til. Hvis det allerede er til stede, legg til navnet i listen (ordbok-verdi).
- Resultaten av denne funksjonen skal returneres, og se slik ut:

```
{'C': ['Anna', 'Paal'], 'E': ['Joakim', 'Ine'], 'A': ['Louise', 'Vegard'], 'F': ['Lasse', 'Kaja', 'Ola'], 'B': ['Gry'], 'D': ['Odd', 'Sven']}
```

b)

Lag en prosedyre `skriv_ut_sortert()` som tar inn karakter ordbøken laget av `sorter_etter_karakter()` og skriver ut denne ordbøken sortert:

```
A : ['Louise', 'Vegard']  
B : ['Gry']  
C : ['Anna', 'Paal']  
D : ['Odd', 'Sven']  
E : ['Joakim', 'Ine']  
F : ['Lasse', 'Kaja', 'Ola']
```


c)

Lag en funksjon `hent_vanligste_karakter()` som tar inn karakter ordbøken og returnerer karakteren som de fleste studenter fikk, og skriv ut resultatet (*Hint: bruk funksjonen `len()` for å se hvor mange studenter fikk karakteren*).

d)

Kall hver funksjon og prosedyre for å sjekke at de fungerer