

Løsningsforslag eksamen DAT120 høst 2021

Oppgave 1 – Flervalgsoppgave

Denne oppgaven er automatisk rettet. Svar i **fet** tekst

- Hva slags språk er Python? **Tolket**
- Hva er datatypen til variabelen «tall» etter tilordningen over? **Int**
- Hva er datatypen til variabelen "innverdi" etter å ha kjørt Python kodelinja over? **Str**
- Hva blir skrevet ut hvis du kjører Python-koden over? **5**
- Hva blir skrevet ut i den første av de to print-setningene i koden over? **3**
- Hva blir skrevet ut i den andre av de to print-setningene i koden over? **13**
- Hva blir skrevet ut hvis du kjører Python-koden over? **None**. Årsak: Funksjonen har ikke noen return setning, så den returnerer ingenting, altså None. Resultat-variabelen inne i funksjonen er en lokal variabel og derfor forskjellig fra Resultat-variabelen utenfor.
- Hva blir skrevet ut hvis du kjører Python-koden over? **22**

Oppgave 2

Denne oppgaven er automatisk rettet

- Hva er 107 i det binære tallsystemet (3%)? **1101011**
- Hva er det binære tallet 100110 i titallssystemet (3%)? **38**

Oppgave 3

```
import math
```

```
a = float(input("Skriv inn den lengste akse, a: "))
b = float(input("Skriv inn den korteste akse, b: "))
resultat = math.pi*a*b
print(f"Arealet for ellipsen med akser {a:.2f} og {b:.2f} er {resultat:.2f}")
```

Merk: Denne likner ganske mye på et av eksemplene fra undervisningen som studentene hadde tilgang til under eksamen, «areal av en sirkel». Har studentene bare kopiert denne skal de bare få ett poeng for å ha skjønnt hva de skal ta utgangspunkt i. De må endre dette eksemplet til å gjøre det oppgaven ber om og ikke mer for å få full score. Oppgaven er ment som en plankeoppgave.

Oppgave 4

```
tall = 10          # Kan være hva som helst som er større enn 0
max = 0
min = -1
sum = 0
antall = 0
while tall > 0:
    tall = float(input("Skriv inn et tall"))
    if tall <= 0:
        break
    if min == -1:
```

```

        min = tall
    if tall < min:
        min = tall
    if tall > max:
        max = tall
    sum += tall
    antall += 1
gjennomsnitt = sum/antall
print(f"Minimum: {min}")
print(f"Maksimum: {max}")
print(f"Gjennomsnitt: {gjennomsnitt}")

```

Oppgave 5

```
import matplotlib.pyplot as plt
```

```
# Utdelt kode
```

```
def funksjonen(tall):
    return tall**2 - tall*500
```

```
# Løsning som bruker list comprehension
```

```
x_koordinater = range(-1000, 1001)
y_koordinater = [funksjonen(x_koordinat) for x_koordinat in x_koordinater]
plt.plot(x_koordinater, y_koordinater, "o-")
plt.show()
```

```
# Løsning som bruker for-løkke. Begge disse skal gi full score.
```

```
x_koordinater = range(-1000, 1001)
y_koordinater = list()
for x in x_koordinater:
    y_koordinater.append(funksjonen(x))
plt.plot(x_koordinater, y_koordinater, "o-")
plt.show()
```

Oppgave 6

```
tabellstorrelse = int(input("Tabellstorrelse: "))
print(" ", end="")
for tall in range(1, tabellstorrelse+1):
    print(f"{tall:4d}", end="")
print()
print("----"*(tall+1))
for tall1 in range(1, tabellstorrelse+1):
    print(f"{tall1:2d}: ", end="")
    for tall2 in range(1, tabellstorrelse+1):
        resultat = tall1*tall2
        print(f"{resultat:4d}", end="")
    print()
```

Merk: Det er ikke så viktig at antall spaces, bindestreker og siffer i print-setningene er riktig. Det som er viktig er at den nøstete for-løkka er formulert riktig og at den skriver ut riktig tall i riktig rekkefølge på riktig linje.

Oppgave 7

```
# Oppgave a)
with open("sykkelpasseringer.txt") as fila:
    fila.readline() # Leser og forkaster første linje, som er
kolonneheadings
    sum = 0
    for linje in fila:
        kolonner = linje.split(";")
        antall_streng = kolonner[3].strip() # Lite viktig, ikke trekk
mye hvis denne mangler
        antall = int(antall_streng)
        sum += antall
    print(f"Totalt antall passeringer: {sum}")

# Oppgave b)
with open("sykkelpasseringer.txt") as fila:
    maalestasjon = input("Skriv inn navnet på målestasjonen: ")
    fila.readline() # Leser og forkaster første linje, som er
kolonneheadings
    sum = 0
    for linje in fila:
        kolonner = linje.split(";")
        if kolonner[1] == maalestasjon:
            antall_streng = kolonner[3].strip() # Lite viktig, ikke trekk
mye hvis denne mangler
            antall = int(antall_streng)
            sum += antall
    print(f"Totalt antall passeringer for stasjonen {maalestasjon}: {sum}")

# Oppgave c)
with open("sykkelpasseringer.txt") as fila:
    maalestasjon = input("Skriv inn navnet på målestasjonen: ")
    maaned = int(input("Skriv inn månedsnummer til måneden, et tall fra 1
til og med 12:"))
    fila.readline() # Leser og forkaster første linje, som er
kolonneheadings
    sum = 0
    for linje in fila:
        kolonner = linje.split(";")
        if kolonner[1] == maalestasjon:
            dato = kolonner[2].split("-")
            maaned_denne = int(dato[1])
            if maaned == maaned_denne:
                antall_streng = kolonner[3].strip() # Lite viktig, ikke
trekk mye hvis denne mangler
                antall = int(antall_streng)
                sum += antall
    print(f"Totalt antall passeringer for stasjonen {maalestasjon} i maaned
{maaned}: {sum}")

# Oppgave d).
with open("sykkelpasseringer.txt") as fila:
    maalestasjon = input("Skriv inn navnet på målestasjonen: ")
    maaned = int(input("Skriv inn månedsnummer til måneden, et tall fra 1
til og med 12:"))
    fila.readline() # Leser og forkaster første linje, som er
kolonneheadings
    sum = 0
```

```

for linje in fila:
    kolonner = linje.split(";")
    if len(kolonner) < 4:
        continue
    try:
        if kolonner[1] == maalestasjon:
            dato = kolonner[2].split("-")
            maaned_denne = int(dato[1])
            if maaned == maaned_denne:
                antall_streng = kolonner[3].strip() # Lite viktig,
ikke trekk mye hvis denne mangler
                antall = int(antall_streng)
                sum += antall
            except ValueError: # Hopper til neste linje i fila hvis
dato[1] ikke kan konverteres til et tall
                pass
            except IndexError: # Hopper til neste linje i fila hvis
dato[1] ikke fins.
                pass
    print(f"Totalt antall passeringer for stasjonen {maalestasjon} i maaned
{maaned}: {sum}")

```

Merk at en student har svart på alle fire deloppgavene ved bare å skrive «oppgave d» svaret, så det er egentlig alt som kreves.

Oppgave 8

```

# Oppgave a)
class Rom:
    def __init__(self, type, kapasitet, romnummer):
        self.type = type
        self.kapasitet = kapasitet
        self.romnummer = romnummer

    # Oppgave c)
    def __str__(self):
        return f"Rom: {self.romnummer} av romtype {self.type} med kapasitet
{self.kapasitet}"

# Oppgave b)
def finn_aktuelle_rom(romliste, minimumskapasitet):
    resultat = list()
    for rom in romliste:
        if rom.kapasitet >= minimumskapasitet:
            resultat.append(rom)
    return resultat

```