IN1000

Disclaimer

Dette er en ikke "offisiell" oversikt over *alt* pensum som kan komme på eksamen, kun en oversikt over noen ting jeg tenkte var nyttig å ha med. Det vil si at dette ikke er ment å være en utømmende liste over alt dere kan få, eller noen tips om hva som kommer på eksamen. Jeg vet ikke hva som kommer på eksamen.

1. Variabler og enkle operasjoner

```
a = 5
b = a > 2        # b blir True
a += 10          # a = a + 10, a blir 15
a *= 2           # a = a * 2
```

2. If-setninger

```
# Enkel if
if a > 5:
    print("Større enn 5")

# If-elif-else
if varepris > 1000:
    return varepris
elif varepris >= 500:
    return varepris + 50
else:
    return varepris + 80

# Logiske operatorer
if a == 3 or b:
    print("Sant")

if "katt" in dyreliste and "hund" in dyreliste:
    return True
```

3. Løkker

While-løkke

```
x = 5
y = 1
while x >= y:
    x = x - y
    y = y * 2
```

```
print(y - x)  # Output: 2

# While med input-validering
valg = -1
while valg < 0 or valg >= len(liste):
    valg = int(input("Skriv et heltall: "))
```

For-løkke

```
# For med range
total = 0
for b in range(5):      # 0, 1, 2, 3, 4
    total += b
print(total)            # Output: 10

# range(start, slutt)
for i in range(1, len(s)):
    print(i)            # 1, 2, 3, ..., len(s)-1

# Iterere over liste
for element in liste:
    print(element)

# Nøstet for-løkke (3 terningkast)
for t1 in range(1, 7):
    for t2 in range(1, 7):
        for t3 in range(1, 7):
            print([t1, t2, t3])
# Output:

# Eksempel på hvordan tre nøstede for løkker med range() kan se ut
for t1 in range(3):
    for t2 in range(3):
        for t3 in range(3):
            print(t1, t2, t3)
# Output:

0 0 0
0 0 1
0 0 2
0 1 0
0 1 1
0 1 2
0 2 0
...       # fjernet noen outputs for å korte ned litt
2 1 1
2 1 2
2 2 0
2 2 1
2 2 2
```

4. Lister

```
# Opprette liste
liste = [1, 2, 3]
tom_liste = []

# Legge til elementer
liste.append(4)         # [1, 2, 3, 4]

# Aksessere elementer
første = liste[0]       # 1
siste = liste[-1]       # 4 (-1 gir siste indeksen i listen)

# Lengde
antall = len(liste)     # 4

# Sjekke om element finnes
if "eple" in liste:
    print("Fant eple")

# Fjerne fra liste (lage en ny liste med alle elementer som ikke er i
listen utsolgte)
nyliste = []
for vare in handleliste:
    if not vare in utsolgte:
        nyliste.append(vare)

# Ta annenhvert element av to forskjellige lister
venner = ["Jonas", "Victor", "Henning"]
fiender = ["Oblig", "Skatteetaten", "Fortnite"]
bordliste = []

for i in range(len(introverte)):
    bordliste.append(introverte[i])
    bordliste.append(ekstroverte[i])
# Output print(bordliste): ["Jonas", "Oblig", "Victor", "Skatteetaten",
"Henning", "Fortnite"]

# Nøstet liste (liste av lister)
tall_lister = [[1, 10], [5, 6], [2, 3]]
for indre_liste in tall_lister:
    for tall in indre_liste:
        print(tall)
        # output: printer alle tallene en om gangen
```

5. Ordbøker (Dictionaries)

```python
# Opprette ordbok
ordbok = {1: 3, 6: 5, 2: 7}
tom_ordbok = {}

# Hente verdi med nøkkel
verdi = ordbok[1]       # 3

# Legge til / oppdatere
ordbok[4] = 9
ordbok[1] += 1          # ordbok[1] gir 3, også øker vi den med 1, blir 4

# Sjekke om nøkkel finnes
if "Brann" in tabellplass_ordbok:
    print("Funnet") #printer funnet hvis "Brann" er en nøkkel i ordboken

# hvis variabelen "nøkkel" ikke er en nøkkel i ordboken så legges den til
med verdi 0
if nøkkel not in ordbok:
    ordbok[nøkkel] = 0

# Iterere gjennom ordbok
total = 0
ordbok = {1: 3, 6: 5, 2: 7}
for k in ordbok:                    # k er nøkkel
    if ordbok[k] < k:
        total += k
# Output: total = 6

# Finne nøkkel basert på verdi
for lag in tabellplass_ordbok:
    if tabellplass_ordbok[lag] == 1:
        return lag
        # returnerer den første nøkkelen som har verdien 1

# Ordbok med lister som verdier
grupper = {}
personers_interesse = {"Per": "Mat", "Palle": "Film", "Espen": "Mat"}
for person in personers_interesse:
    interesse = personers_interesse[person]
    if not interesse in grupper:
        grupper[interesse] = []
    grupper[interesse].append(person)
# Output: {"Mat": ["Per", "Espen"], "Film": ["Palle"]}

# .keys(), .values() og .items()
# Eksempel ordbok:
ordbok = {
    "eple": "apple",
    "banan": "banana",
    "appelsin": "orange"
}

# Henter og printer alle nøklene
```

```
for k in ordbok.keys():
    print(k)
# Output:
eple
banan
appelsin

# Henter og printer alle verdiene
for v in ordbok.values():
    print(v)
# Output:
apple
banana
orange

# Henter og printer både nøkler og verdier
for nøkkel, verdi in ordbok.items():
    print(nøkkel, "og", verdi)
# Output:
eple og apple
banan og banana
appelsin og orange
```

6. Strenger

```
# Lengde
s = "kokos"
lengde = len(s)         # 5, siden "kokos" består av 5 bokstaver

# hente en enkelt bokstav i et ord
første_tegn = s[0]      # "k"
andre_tegn = s[1]       # "o"

# Iterere over streng
s = "ifi er fint"
for bokstav in s:
    print(bokstav)
# Output:
i
f
i

e
r

f # legg merke til at mellomrommene blir printet også over her
i
n
t
```

```
# gjøre en setning om til en liste av ordene
setning = "jeg digger film"
ord_liste = setning.split()     # ["jeg", "digger", "film"]


# String concatenation med liste av ord
ord_liste = ["ifi", "er", "et", "fint", "sted"]
setning_uten_mellomrom = ""

for ord in ord_liste:
    setning_uten_mellomrom = setning_uten_mellomrom + ord

print(setning_uten_mellomrom)
# Output:
# ifieretfintsted


# .lower() og .upper()
tekst = "HALLO"
liten = tekst.lower()           # "hallo"

# Sjekke om substring finnes
if "kat" in "kattepus":
    print("Fant") # output: "Fant", kat finnes i strengen den sjekker
```

7. Funksjoner

```
# Enkel funksjon
def si_hei():
    print("Hei!")

# Med parameter
def si_hei(navn):
    print("Hei", navn)

# Med retur-verdi
def legg_sammen(a, b):
    return a + b

resultat = legg_sammen(3, 5)    # 8

# Funksjon som kaller annen funksjon
def poeng(tekst):
    if tekst == "lyn":
        return 0
    else:
        return len(tekst)

def poengsum(liste):
    total = 0
    for tekst in liste:
```

```
            total += poeng(tekst)
        return total
```

8. Klasser og Objekter

Enkel klasse

```
class Person:
    def __init__(self, alder):
        self._alder = alder

    def bursdag(self):
        self._alder += 1

    def hent_alder(self):
        return self._alder

# Opprette objekt
per = Person(5)
per.bursdag()
alder = per.hent_alder()    # 6
```

Klasse med liste

```
class Kategori:
    def __init__(self, navn):
        self._navn = navn
        self._plagg = []

    def nytt_plagg(self, farger):
        self._plagg.append(Plagg(farger)) # Lager et nytt Plagg() objekt
og legger i listen

    def finn_plagg_med_farge(self, farge):
        resultat = []
        for plagg in self._plagg:
            if plagg.har_farge(farge):
                resultat.append(plagg)
        return resultat
```

Klasse som refererer til annen klasse

```
class Barnehage:
    def __init__(self, barn):
        self._barn = barn       # Person-objekt
```

```
    def bursdag(self):
        self._barn.bursdag()    # Kaller metode på Person

    def bytte(self, nytt_barn):
        self._barn = nytt_barn

# Bruk
per = Person(2). # per blir her et objekt av klassen Person()
maurtua = Barnehage(per)
maurtua.bursdag()               # per blir 3 år
```

9. Filhåndtering

```
# Lese fra fil
fil = open("hotellnavn.txt", "r")
for linje in fil:
    info = linje.strip().split()    # Fjern whitespace og del opp
    rom_nr = int(info[0])           # Konverter til int
    ant_senger = int(info[1])
    fasiliteter = []
    for fasilitet in info[2:]:      # Lager en liste med alle elementer
fra index 2 -> slutten av listen
        fasiliteter.append(fasilitet.lower())
fil.close()

# Eksempel fil-innhold:
# 101 3 balcony wifi
# 102 2 wifi
```

Telle forekomster

```
# Med ordbok
dyreliste = ["hund", "katt", "hund", "mus", "katt", "hund", "fugl"]
dyrebok = {}

for dyr in dyreliste:
    if dyr in dyrebok:
        dyrebok[dyr] += 1
    else:
        dyrebok[dyr] = 1

print(dyrebok)
# Output:
# {'hund': 3, 'katt': 2, 'mus': 1, 'fugl': 1}

# Med .count()
liste = [1, 2, 3, 2, 2, 1, 4, 5, 2]
element = 2
telling = liste.count(element)
```

```
print("Antall", element, "i listen:", telling)

# Output:
# Antall 2 i listen: 4
```

Finne største tall

```
størsteTall = liste[0]
for element in liste:
    if element > størsteTall:
        størsteTall = element
```

Sjekke om alle i liste oppfyller krav

```
alle_ok = True
for element in liste:
    if not element.oppfyller_krav():
        alle_ok = False
return alle_ok
```

Bygge ny liste basert på betingelse

```
navn_liste = ["Ada", "Ola", "Kari", "Per", "Emma", "Jonas"]
lange_navn = []

for navn in navn_liste:
    if len(navn) > 3: #sjekker om navnet har mer enn 3 bokstaver
        lange_navn.append(navn)

print(lange_navn)
# Output:
# ['Kari', 'Emma', 'Jonas']
```

Nøstede strukturer - ordbok med lister

```
# Fra ordbok {person: interesse} til {interesse: [personer]}
grupper = {}
for person in personer_interesse:
    interesse = personer_interesse[person]
    if interesse not in grupper:
        grupper[interesse] = []
    grupper[interesse].append(person)
```

11. Nyttige Funksjoner

```
len(liste)              # Lengde av liste/streng
range(5)                # 0, 1, 2, 3, 4
range(1, 5)             # 1, 2, 3, 4
int("42")               # Konverter streng til int
str(42)                 # Konverter int til streng
input("Tekst: ")        # Les input fra bruker (returnerer streng)
min(liste)              # Minste verdi
max(liste)              # Største verdi
liste.count(element)    # Hvor mange ganger element er i liste
set(liste)              # Lager set (unike elementer)
```

12. Vanlige Feil å Unngå

```
# Husk å konvertere input til int hvis den skal være det!
alder = input("Alder: ")        # Dette er en streng!
alder = int(input("Alder: "))   # RIKTIG


# Husk å bruke == istedenfor = når man skal sammenligne
if x = 5:                       # FEIL
if x == 5:                      # RIKTIG
```