Mentimeter

IN1000 Gruppetime Uke 4

Gruppe 4 - Design

Mentimeter

Plan for i dag

- Kort oppvarming om løkker
- Gjennomgang av pensum:
  - for-løkker
  - while-løkker

Mentimeter

Fagordsquiz: looper i Python

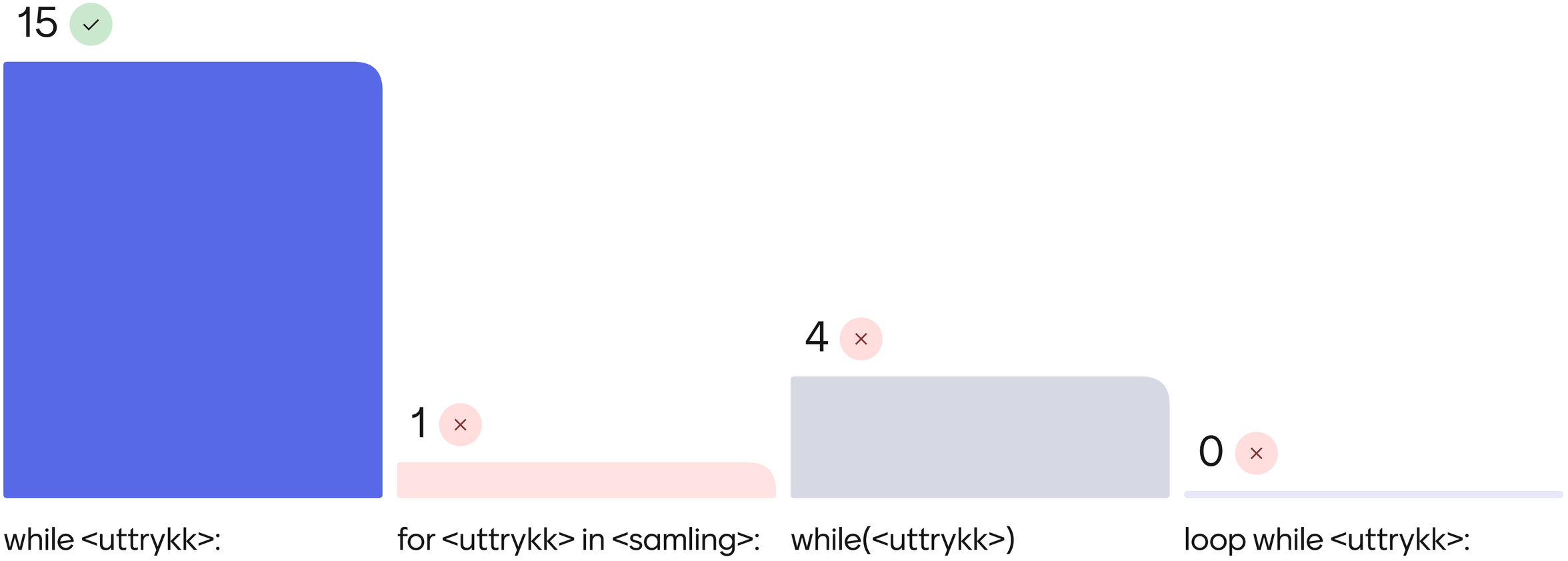
- Kort quiz om sentrale begreper og syntaks
- Ingen poengscore
- Ingen tidspress

1

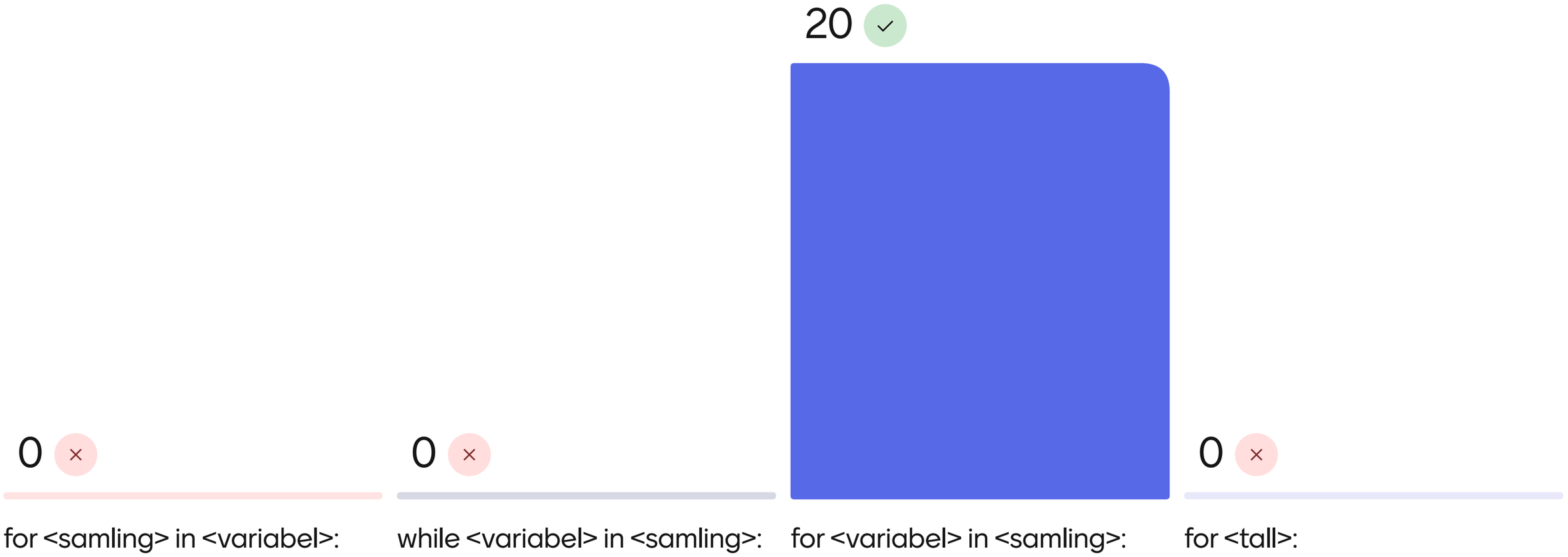
Hva vil det si å *iterere*?



Hvordan lager man en while-løkke?

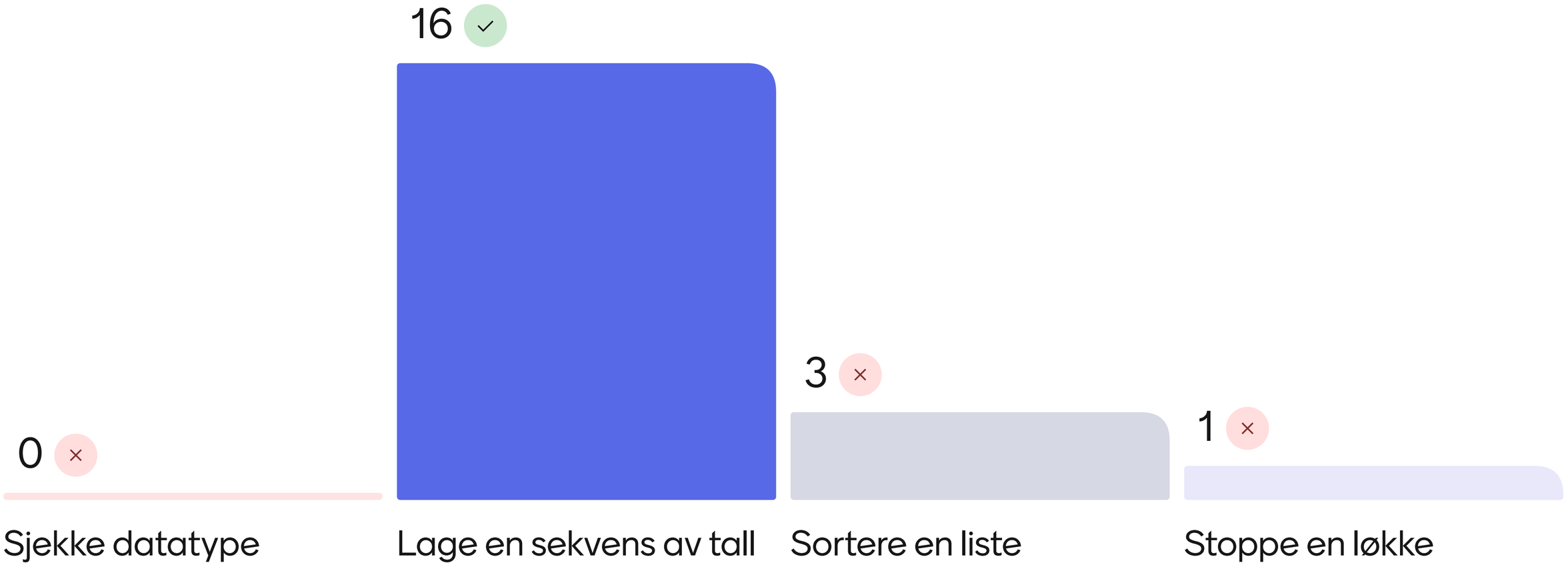


Hvordan lager man en for-løkke?



Mentimeter

Hva brukes range(<tall>) ofte til?



Mentimeter

Hva er løkker i programmering

- Løkker har to hovedfunksjoner
  - De lar oss repetere kode
  - De lar oss iterere gjennom samlinger og andre objekter
- For-løkker itererer gjennom et objekt
- While-løkker kjører så lenge en betingelse er sann

```
for i in range(5):
    print("Hei")

tall = 0
while tall != 5:
    tall += 1
    print("Hei")
```

Mentimeter

While-løkker

- Fungerer som en if-setning som sjekkes flere ganger:
  - Hvis beslutningen stemmer så fortsetter vi innholdet
  - Hvis beslutningen ikke stemmer så stopper vi
- Nyttig hvis vi ønsker å repetere kode en ukjent antall ganger

- Eksempel:
  - En while-løkke som kjører helt til bruker skriver "stopp"

```
gjesteliste = []

svar = input("Hva er navnet til gjesten (skriv stopp for å stoppe)?: ")

while svar != "stopp":
    gjesteliste.append(svar)
    svar = input("Hva er navnet til gjesten (skriv stopp for å stoppe): ")

print(gjesteliste)
```

Mentimeter

Typisk feil med while-løkker

- Det er nå man muligens kommer til å oppleve evige løkker (kode som aldri stopper)
- Pass på at løkken faktisk terminerer til slutt
- ctrl + c for å terminere programmet tidlig

1

Hva blir printet ut?

9

4

2

4-10

4 5 6 7 8 9 10

9

9

4 5 6 7 8 9 10

```
x = 3
while x < 10:
    x += 1
    print(x)
```

Hva blir printet ut?

9

45678910

4 5 6 7 8 9 10
11

4, 5, 6, 7, 8, 9

45678910

11

4

4 5 6 7 8 9 10

```
x = 3
while x < 10:
    x += 1
    print(x)
```

Hva blir printet ut?

3 4 5 6 7 8 9 10

4 5 6 7 8 9 10

4

4 5 6 7 8 9

```
x = 3
while x < 10:
    x += 1
    print(x)
```

4, 5, 6, 7, 8, 9

Hva er a og b for hver iterasjon. Hva blir printet ut?

8, 4, 0

a = 8 b = 2

A = 8 B = 2

a=-4 b=8

a = 8, b = 2

a = -4 b = 8

1 a = 8 1 b = 2
2 a = 4 2 b = 4
3 a = -4 3 b = 8

a = 8 b =2, a = 4 b = 4, a = -4 b = 8

```
a = 10
b = 1

while a > 0:
    b = b * 2
    a = a - b

print("a =", a)
print("b =", b)
```

Hva er a og b for hver iterasjon. Hva blir printet ut?

b = 2 a = 10-1
a = 9 b = 2

b = 2, a = 8. b = 4, a = 4. b = 8, a = -4 (den blir ikke sann, så den blir ikke printet)

b = 2 a = 8 b = 4 a = 4 b = 8 a = -4 Dette printes: b=8 & a=-4

-4, 10 8, 1

```
a = 10
b = 1

while a > 0:
    b = b * 2
    a = a - b

print("a =", a)
print("b =", b)
```

a = -4, b = 8

b = 2 a = 9 b = 4 a = 7 b = 8 a = 3 ???

a=0, b=2

a= 8, 4, 0 b= 2, 4, 8

Hva er a og b for hver iterasjon. Hva blir printet ut?

a = -4 b = 8

a = 0 b = 10

a = 8 b = 2 a = 6 b = 4 a = 2 b = 8 a = -6 b = 16

```
a = 10
b = 1

while a > 0:
    b = b * 2
    a = a - b

print("a =", a)
print("b =", b)
```

Hva blir printet ut?

1, 3, 5, 7, 9

5

1, 3, 5, 7, 9

1, 3, 5, 7, 9

9

3, 5, 7, 9, 11

3,5,7,9,11

1, 3, 5, 7, 9, 11 osv (programmet slutter aldri?)

```
tall = 1

while tall != 10:
    print(tall)
    tall += 2
```

Hva blir printet ut?

1 3 5 7 9 11 13 15.....

1, 3, 5, 7, 9,

3, 5, 7, 9, 11, 13, ... Uendelig

1, 3, 5, 7, 9, 11, 13, 15 etc

Uendelig

1, 3, 5, 7, 9, 11, 13 osv

1, 3, 5, 7, 9

1 3 5 7 9 11 osv.osv.

```
tall = 1

while tall != 10:
    print(tall)
    tall += 2
```

Hva blir printet ut?

1357911uendelig

Går i uendelig

```
tall = 1

while tall != 10:
    print(tall)
    tall += 2
```

Mentimeter

For-løkker

- Nyttig når vi skal:
  - Kjøre noe et eksakt antall ganger
  - Iterere gjennom en samling
  - Kan også iterere gjennom andre objekter som strenger

```
liste = [1, 2, 3]

for item in liste:
    print(item) #Vil printe ut 1, 2, 3

streng = "Hei"

for char in streng:
    print(char) #Vil printe ut "H", "e", "i"
```

I eksemplene til høyre så er item og char en variabel som peker på hvert element i samlingen/strengen

item og char er bare variabler, så er opp til deg hva du ønsker å kalle dem

Mentimeter

For-løkker og range()

- Vi kan gjøre "for i in range(x)" for å kjøre noe x ganger
- range() er et objekt som ligner på en liste med stigende eller minkende tall

```
for i in range(5):
    print(i) #Dette vil printe ut 0, 1, 2, 3, 4

range(5) #Fra og med 0 opp til 5 (inkluderer ikke 5)
range(3) #Fra og med 0 opp til 3 (inkluderer ikke 3)
range(0, 10, 2) #Fra og med 0 opp til 10 (i steg av 2)
print(list(range(5))) #Dette vil printe ut [0, 1, 2, 3, 4]
print(list(range(0, 10, 2))) #Dette vil printe ut [0, 2, 4, 6, 8]
```

Mentimeter

Typisk feil med for-løkker

- Ikke god kodeskikk å legge til eller fjerne elementer i en samling man itererer over
- En for-løkke holder på en indeks og fortsetter å øke den basert på lengden til listen
- Koden til høyre skal fjerne alle partall. Hva skjer galt?
- "if t % 2 == 0" betyr "hvis dette tallet er delelig med 2"

```
tall = [1, 2, 4, 5, 6]

for t in tall:
    #print(t)
    if t % 2 == 0:
        tall.remove(t)

print(tall)
```

Mentimeter

Oppgave 1

Print ut en firkant til terminalen. Med bredde og lengde *n*.
Man kan for eksempel bruke "*" for å tegne firkanten

Gjerne prøv å lag en hul og ikke hul versjon

Eksempel output til høyre



Oppgave 2

Print ut en rettvinklet trekant til terminalen. Med bredde og lengde *n*.

Gjerne prøv å lag en hul og ikke hul versjon

Eksempel output til høyre



Oppgave 3

Print ut en pyramide til terminalen. Med bredde og lengde *n*.

Gjerne prøv å lag en hul og ikke hul versjon

Eksempel output til høyre

