1) a) Ved bruk av python

Standardavvik $\sigma = 0.106$

b) $KI = \bar{x} \pm z_{\alpha/2} \cdot \frac{\sigma}{\sqrt{n}}$

$[20.438, 20.532]$

c) Prediksjonsintervall

$[20.257, 20.713]$

2) a) $[20.298, 20.322]$

98% konfidensintervall

b) Instrumentenes intervall overlapper ikke.

Dette tyder på at instrumentene ikke er enige om temperaturen i rommet. Mulig kalibreringsfeil.

c) to målinger

$[20.321, 20.439]$, ikke andre

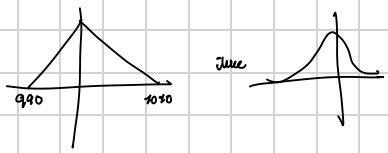
3) a) mean: 1010 Ω

min: 990 Ω

σ: 5.77 Ω

b) 0.58%

c)



Ikke

d) 0.58 %

Kode:

```
import numpy as np


#Oppgave 3

#Oppgave 3 a) Maks, min og standardavvik på R

R = 1000

R_plus = R * 1.01

R_min = R * 0.99

R_std = np.std([R_min, R_plus])



print(f"Oppgave 3, deloppgave a)")

print(f"Maks verdi R: {round(R_plus, 3)} Ω")

print(f"Minuns verdi R: {round(R_min, 3)} Ω")

print(f"Standardavvik R: {round(R_std, 3)} Ω")

print(f"\n")


#Oppgave 3 b) Relativt standardavvik

s_R = (R_plus - R_min)/np.sqrt(12) #Standardavvik for uniform fordeling

m_R = R

RS = s_R / m_R

print(f"Oppgave 3, deloppgave b)")

print(f"Relativ standardavvik: {round(RS * 100, 2)}%")

print(f"\n")


#Oppgave 3 d) Standaravvik

delta_r = 10

r_values = 1000 + 2*delta_r * (np.random.rand(100000) - 0.5)

relative_std = np.std(r_values, ddof=1)/np.mean(r_values)
```

```
print(f"Oppgave 3, deloppgave d)")



print(f"Relativ standardavvik: {round(relative_std * 100, 2)}%")



import numpy as np

#Oppgave 1

maalinger = [20.6, 20.4, 20.4, 20.6, 20.4, 20.8, 20.5, 20.5, 20.5, 20.4, 20.5, 20.5, 20.5, 20.5, 20.4, 20.4, 20.4, 20.5, 20.3, 20.6]

n = len(maalinger)

#Oppgave 1 a) Standardavvik

sigma = np.std(maalinger)

print(f"Standardavvik: {round(sigma, 3)}")

#Oppgave 1 b) 95% konfidensintervall

x_bar = np.mean(maalinger)

z = 1.96 #95%

s = sigma

CI_plus = x_bar + (z * (s / np.sqrt(n)))

CI_minus = x_bar - (z * (s / np.sqrt(n)))

print(f"95% konfidensintervall: [{round(CI_minus, 3)}, {round(CI_plus, 3)}]")

#Oppgave 1 c) 95% Prediksjonsintervall

m_T = x_bar

s_T = sigma

t_p = 2.093 #Fra tabell, n = 20, p = 0.975

PI_plus = m_T + (t_p * s_T * np.sqrt(1 + (1 / n)))
```

```
PI_minus = m_T - (t_p * s_T * np.sqrt(1 + (1 / n)))

print(f"95% prediksjonsintervall: [{round(PI_minus, 3)}, {round(PI_plus, 3)}]")



import numpy as np

#Oppgave 2
maalinger = [20.4, 20.4, 20.4, 20.2, 20.4, 20.3, 20.4, 20.5, 20.4, 20.4, 20.4, 20.4, 20.1, 20.3, 20.3, 20.2, 20.3, 20.2, 20.3, 20.3]
n = len(maalinger)

#Oppgave 2
#Oppgave 2 a) 95% konfidensintervall
x_bar = np.mean(maalinger)
z = 1.96 #95%
s = np.std(maalinger)

CI_plus = x_bar + (z * (s / np.sqrt(n)))
CI_minus = x_bar - (z * (s / np.sqrt(n)))

print(f"95% konfidensintervall med 20 målinger: [{round(CI_minus, 3)}, {round(CI_plus, 3)}]")

#Oppgave 2 c)
maalinger_ti = maalinger[0:10]
n_ti = len(maalinger_ti)

x_bar_ti = np.mean(maalinger_ti)
z_ti = 1.96 #95%
s_ti = np.std(maalinger)

CI_plus_ti = x_bar_ti + (z_ti * (s_ti / np.sqrt(n_ti)))
CI_minus_ti = x_bar_ti - (z_ti * (s_ti / np.sqrt(n_ti)))
```

```
print(f"95% konfidensintervall med 10 målinger: [{round(CI_minus_ti, 3)}, {round(CI_plus_ti, 3)}]")
```