

🎲 Terningkast og statistikk - del 1: To terninger

LØSNINGSFORSLAG

7.-9. trinn · 35 min · SkapLab – Python

Hva du skal lære

Simulere kast med to terninger og finne sumfordelingen. Forstå at ikke alle summer er like sannsynlige – og bevise det med kode.

Kodeeksempel

```
import random

antall = 10000
teller = {} # ordbok for å telle hvert utfall

for i in range(2, 13):
    teller[i] = 0 # initialiser alle mulige summer

for _ in range(antall):
    t1 = random.randint(1, 6)
    t2 = random.randint(1, 6)
    teller[t1 + t2] += 1

print("Sum | Antall | Prosent | Graf")
for sum_verdi in range(2, 13):
    ant = teller[sum_verdi]
    pst = ant / antall * 100
    bar = "█" * int(pst)
    print(f" {sum_verdi:2} | {ant:5} | {pst:5.1f} % | {bar}")
```

Oppgaver

1. Kjør koden. Hvilken sum har høyest frekvens?
2. Øk til 100 000 kast. Endrer fordelingen seg nevneverdig?
3. Finn teoretisk sannsynlighet: hvor mange kombinasjoner gir sum 7?
4. Legg til en tredje terning og finn ny fordeling.
5. Finn gjennomsnittet av alle kast og sammenlign med teorien (7,0).
6. Utfordring: lag en ordbok for to terninger der nøkkelen er (t1, t2).

Løsningsforslag - gjennomsnitt og teoretisk verdi

```
import random

antall = 100000
teller = {i: 0 for i in range(2, 13)}
total = 0

for _ in range(antall):
    s = random.randint(1,6) + random.randint(1,6)
    teller[s] += 1
    total += s

snitt = total / antall
print(f"Gjennomsnitt: {snitt:.3f} (teoretisk: 7.000)")

mest = max(teller, key=teller.get)
print(f"Vanligste sum: {mest} ({teller[mest]/antall*100:.1f} %)")
```

max(ordbok, key=ordbok.get) returnerer nøkkelen med høyest verdi i en ordbok. Svært nyttig for å finne det hyppigste utfallet.