

# Teori spørsmål – Obligatorisk innlevering 2

## Teorioppgave 1 - if-test/if-statement

«Forklar hva en if-test/if-statement er med egne ord»

- En **If**-statement er en lås som stopper deg fra å kjøre underliggende kode med mindre du har nøkkelen. Men andre ord, du kan sette opp et(eller flere) krav som må møtes før underliggende kode and kjøres.

«Beskriv også sammenhengen mellom if, elif og else. »

- **If** betyr at noe skjer hvis et krav er møtt.
- **Else**, betyr at hvis ikke **If**-kravet ble møtt så kan koden dens kjøres i stedet. Hvis du har flere **If/elif**-statements, så må ingen av deres krav være oppfylt for at **else** skal kunne kjøre underliggende kode.
- **Elif**, er en forkortelse på **else if**, og er en blanding av de to andre logikk-gatene over. Den betyr at hvis ikke **If**-kravet ble møtt, OG at sitt eget krav er oppfylt, så kan du kjøre dens underliggende kode.

## Teorioppgave 2 - liste

«Forklar hva lister er i Python og hva de kan brukes til.»

- En liste er et variabel hvor du kan lagre informasjon i form av en liste.
- Du kan lagre alle typer variabler inni en liste.
- De bruker til å holde styr på data, sortere data, eller generelt jobbe med mye data. Det er mye enklere enn å lage 1000 forskjellige variabler.
- Lister kan ha forskjellige dybder.
  - Eks: List = [1, 2, 3] og List = [ [1, 2], [3, 4] ]
  - Note: Dype lister er vridende å kopiere over til andre lister (i Python)

## Teorioppgave 3 - løkker

«Forklar hva "løkker" er, og gi noen eksempler på tilfeller man kan ønske å benytte dem.»

- En løkke er et område i koden som blir gjentatt x antall ganger, eller til et annet krav blir møtt.
- Det finnes flere forskjellige, som for eksempel for-løkker og while-løkker.
- Er veldig nyttig i spill, hvor man ofte må lage mange fiender eller lignende. Da lager man bare kode som lager EN fiende og legger fienden inn i en liste, også dytter vi den koden inn i en loop.
- Er også nyttig i apper, de fleste apper/spill kjøres i en kontinuerlig loop frem til brukeren krysser de ut.