

# INF1000 - Ekstrainnlevering

Temaer på innleveringen: *Hele pensum fram til nå.*

## Oppgave E.1)

**Tema:** *Beslutninger (if/else)*

**Filnavn:** `Rekkefolge.java`

- a) Lag et program som leser inn tre tall fra brukeren.
- b) Sjekk om tallene er sortert i stigende rekkefølge, synkende rekkefølge eller ingen rekkefølge. Avhengig av dette skal programmet printe "Stigende", "Synkende" eller "Ingen".

Eksempel på en kjøring av programmet

```
Skriv inn 3 tall:
```

```
1  
2  
3
```

```
Stigende
```

```
Skriv inn 3 tall:
```

```
3  
2  
4
```

```
Ingen
```

## Oppgave E.2)

**Tema:** *Arrayer*

**Filnavn:** `ArrayLek.java`

- a) Lag en array `int[] heltall` med plass til 5 tall og la brukeren skrive inn tallene i arrayen.
- b) Lag en metode `storsteTall` som tar i mot en array og som finner det største tallet i arrayen. Kall metoden med arrayen fra oppgave a som parameter, ta vare på returverdien i en variabel, og skriv ut denne verdien til terminalen.

- c) Lag en metode `reverserArray` som tar i mot en array og som flytter alle elementene over i en annen array, men i motsatt rekkefølge. Metoden skal returnere den nye arrayen. Print ut elementene i den nye arrayen til terminalen. *Hint:* Her kan det lønne seg å bruke en teller for posisjonen i den gamle arrayen og en teller for posisjonen i den nye arrayen.

### Oppgave E.3)

**Tema:** *For-løkke og String*

**Filnavn:** `Vokaler.java`

- a) Lag et program som leser inn et ord fra brukeren.
- b) Lag en array `vokaler` som inneholder alle de internasjonale vokalene (a, e, i, o, u, y).
- c) Bruk deretter for-løkker til å gå gjennom ordet du leste inn fra brukeren og som teller hvor mange vokaler det er i ordet. Skriv ut resultatet til terminalen. *Hint:* Her må du bruke en for-løkke for å gå gjennom ordet og en for-løkke for å gå gjennom vokaler. Det kan også være lurt å bruke `String` sin metode `charAt`.

Eksempel på en kjøring:

```
Skriv inn et ord:  
Informatikk  
Antall vokaler er 4
```

### Oppgave E.4)

**Tema:** *Innlesing av fil og String*

**Filnavn:** `PrinsessaOgErten.java`

- a) Lag et program som leser inn filen "princessandthepea.txt". Det er kun ett ord per linje. *Hint:* Bruk en while-løkke så lenge det fortsatt finnes linjer i teksten.
- b) Sjekk lengden på hvert ord, og sjekk hvor mange av ordene som har lengde 4. *Hint:* Bruk `String` sin metode `length`.
- c) Lag deretter en metode `forsteBokstav` som tar i mot et ord og en bokstav og sjekke om ordet starter med bokstaven. Hvis det gjør det skal ordet skrives ut i terminalen. Bruk denne metoden til å finne hvor mange ord som begynner på 'a' i filen.

## Oppgave E.5)

**Tema:** *Ordreløkke (while)*

**Filnavn:** MinOrdrelokke.java

- a) Lag en metode `printSum` som leser inn to tall fra brukeren og printer summen av dem til terminalen.
- b) Lag en metode `printHei` som leser inn et navn fra brukeren og printer "Hei, [innlest navn]" til terminalen.
- c) I `main`-metoden: Skriv kode som gjør at programmet leser inn tall fra brukeren helt fram til brukeren taster 0.
- d) Sjekk nå hvilket tall brukeren taster inn. Hvis tallet er 1 skal programmet kalle metoden `printSum` og hvis tallet er 1 skal programmet kalle metoden `printHei`. Dersom det er et annet tall skal programmet gi en egenlaget feilmelding.
- e) Lag også en metoden `printMeny` som printer menyen til brukeren hver gang brukeren skal kunne taste inn et tall.

Eksempelkjøring av programmet (brukeren har valgt menyvalg 1 og deretter 0):

```
Meny:
0. Avslutt
1. Print summen av to tall
2. Print velkomsthilsen
Tast valg:
1
Skriv inn et tall:
2
Skriv inn et tall til:
4
Summen av tallene er 6
Meny:
0. Avslutt
1. Print summen av to tall
2. Print velkomst hilsen
Tast valg:
0
```

## Oppgave E.6

**Tema:** *Klasser og objekter*

**Filnavn:** Oppgave06.java, Person.java og Bil.java

- a) Lag en klasse `Bil` som inneholder instansvariablene `nummerskilt` og `eier`, begge av typen `String`. Lag også metoder for å lagre verdier i variablene og for å hente dem ut.
- b) Lag en annen klasse `Person` som inneholder instansvariablene `navn` av typen `String`, `alder` av typen `int` og `bil` av typen `Bil`.
- c) Lag en egen klasse `Oppgave06` som inneholder en `main`-metode. Les deretter inn et navn og en alder fra brukeren og lag et `Person`-objekt med det innleste navnet og alderen. Lag deretter et `Bil`-objekt hvis den innleste alderen er større enn 18, og lagre det i `Person`-objektet.

## Oppgave E.7

**Tema:** *Klasser og objekter*

**Filnavn:** `Oppgave07.java` og `Brev.java`

- a) Lag en klasse `Brev` som inneholder instansvariablene `mottaker` og `avsender`, begge av typen `String`. Lag metoder for å sette disse variablene. Lag også en array som tar vare på tekststrenger. *Hint:* Arrayen kan for eksempel ha plass til 100 elementer.
- b) Lag en metode i klassen som heter `skrivLinje` og som tar i mot en tekststreng og lagrer den i neste ledige posisjon i tekst-arrayen. *Hint:* Bruk en teller for å holde styr på neste ledige posisjon.
- c) Lag metode `lesBrev` som returnerer brevets innhold tilbake til der det ble kalt fra. Brevet skal være på formen gitt nedenfor. *Hint:* For penere utskrift: Ha en teller på hvor mange linjer du har lagt til i brevet.
- d) Lag deretter en klasse `Oppgave07` som oppretter et `Brev`-objekt. Legg til `mottaker`, `avsender` og minst to linjer i objektet ved hjelp av `skrivLinje`. Kall deretter på `lesBrev`-metoden og print ut resultatet til terminalen.

Formen på brevet som skal leveres:

```
Hei, [mottakers navn]!
[blank linje]
første linje i brevet
andre linje i brevet
....
[blank linje]
Hilsen fra
[avsenders navn]
```

Eksempelkjøring på hvordan programmet kan se ut:

Hei, Espen Askeladd!

Hvordan har du det?  
Jeg har det bare bra!

Hilsen fra  
Per Askeladd

## Frengangsmåte for innleveringer i INF1000

1. Lag en fil som heter README.txt. Følgende spørsmål skal være besvart i filen:
  - Hvordan synes du innleveringen var? Hva var enkelt og hva var vanskelig?
  - Hvor lang tid (ca) brukte du på innleveringen?
  - Var det noen oppgaver du ikke fikk til? Hvis ja:
    - Hvilke(n) oppgave er det som ikke fungerer i innleveringen?
    - Hvorfor tror du at oppgaven ikke fungerer?
    - Hva ville du gjort for å få oppgaven til å fungere hvis du hadde mer tid?
2. Logg inn på [Devilry](#).
3. Lever de alle .java-filene og README.txt i *samme innlevering*.
4. Husk å trykke lever og sjekk deretter at innleveringen din er komplett.

Denne (frivillige) innleveringen er minimum av hva du bør ha programmert i løpet av uken. Du finner flere oppgaver i [Trix](#). Bruk filtre for å velge tema (uke) og/ eller nivå (øvelse/ mengde/ utfordring). Om du logger deg inn og registrerer hvilke du løser, vil du (og bare du) kunne holde rede på andelen oppgaver du har løst for hver uke eller nivå.