

Læringmål: Arrays, File IO, Rutediagram, psudokode.

## 1 Læs fil

En essentiel del af programmering, er at kunne gemme og læse data. Når vi gemmer data, gemmer vi det i en fil. File-ekstention fortæller hvordan filen er struktureret og skal læses. Der findes mange måder at strukturere data på. F.eks. .pdf, .docx, .jpg. Ekstention angiver hvilken type fil det drejer sig om. Det hjælper brugeren til at finde det rigtige program til at åbne filen med. En opgave som i dag er overtaget af operativsystemet. Den filtype vi vil bruge er .txt filer. .txt betyder at filen er gemt i ren tekst(ASCII værdier) og kan åbnes og læses med f.eks notepad.

```
String[] lines = loadStrings("list.txt");
println("there are " + lines.length + " lines");
for (int i = 0; i < lines.length; i++) {
    println(lines[i]);
}
```

Figure 1: funktion som læser linjerne i en fil

For at kunne forstå koden i Figure 1, skal vi kunne forstå den første linje. Kommandoen `loadstring("list.txt")`, læser hele filen `list.txt` og returnerer et array af Strings. Hver enkelt string i arrayet er én linje i filen. Her efter er koden triviell. Vi udskriver antallet af linjer, og herefter benyttes et for-loop til at løbe gennem array'et og udskrive de enkelte linjer.

Læs mere om `loadstring` i [Processing reference](#).

## 2 Skriv fil

Tilsvarende til `loadstring()` tilbyder `processing` `savestring()`. `Savestring()` skal have to parametre. Den første skal være en String, som indeholder navnet på filen. Den anden parameter skal indeholde en array af String, som indeholde de data man vil gemme. Vær dog opmærksom på at findes der en fil med samme navn i forvejen, vil denne blive overskrevet.

```
String names = "Aksel Albert Alfred Anker Anton Arne Asger
Agnes Agnete Alma Almine Ane Anine Anna Anni Asta Astrid Augusta";
// Deler String ved alle mellemrum og flytter data over i et array
String[] list = split(names, ' ');
// Skriver vores array til filen, én string på hver linje.
saveStrings("names.txt", list);
```

Figure 2: funktion som skriver til en fil

Processing tilbyder en mere avanceret funktion til at skrive til filer med. Hvis du har brug for det, kan du kigge på `createWriter()` og `createReader()` funktionerne.

### 3 Opgave

Du skal lave et program som kan læse og skrive til en fil. Brug f.eks `draw.io` eller `lucid.app` til at tegne dine diagrammer. Du kan også bruge Microsoft visio eller andet.

1. Lav et rutediagram over et program som modtager `mouseX` og `mouseY` og gemmer 10 værdisæt i en fil. Læs om rutediagrammer her.
2. Skriv pseudokode efter dit rutediagram. Læs om pseudokode her.
3. Lav programmet efter dit rute diagram.
4. Lav et rutediagram over et program, som kan læse en fil med koordinatsæt og udskriver disse koordinater til skærmen.
5. Skriv pseudokode over dit rutediagram. Læs om pseudokode her.
6. Lav programmet efter dit rute diagram.
7. Ved brug af de teknikker du nu har lært, lav en funktion som kan tilføje data til en fil.