

Personprogram

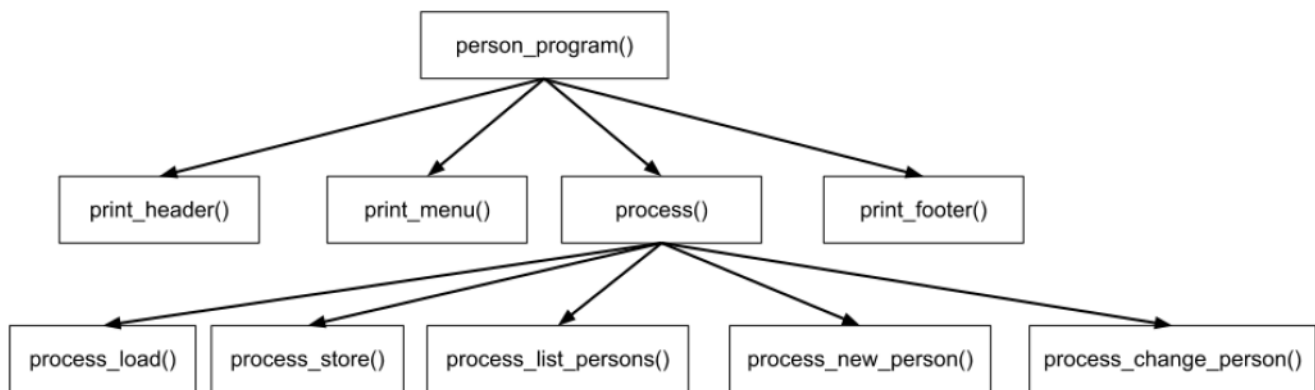
Læringsmål:

- Store programmer
- Betingelser
- While-løkker
- Utskrift

Pensum:

- 3.7 - User-Defined Functions That Return a Single Value
- 4.1 - The if statement
- 4.4 - The switch statement
- 5.1 - The for loop
- 5.3 - While loops
- 6.2 - MATLAB Program Organization

I denne oppgaven skal funksjoner fra [Personstruct](#) og [Persondatabase](#) settes sammen til et kjørende program. Det skal vises en meny til brukeren og la brukeren velge hva den vil gjøre.



Hittil i øvingsopplegget har oppgaveteksten lagt opp til å skrive i en bunn-til-topp (bottom-up) stil. Det vil si å skrive de laveste funksjonen i kalltreet (se figur) først og de øverste til slutt. Den motsatte metoden, topp-til-bunn (top-down) er også mye i bruk, det vil si å skrive de øverste funksjonene først. Dette krever at man holder litt flere tanker i hodet samtidig, men fører etter trening ofte til en bedre løsning.

Definer følgende funksjoner:

```

function P = process (P, choice )
fprintf ('Du har valgt %i\n', choice );
end

function printHeader ()
fprintf ('Skriver ut en velkomstmelding \n');
end

function printMenu ()
fprintf ('Skriver ut en meny \n');
end

function printFooter ()
fprintf ('Skriver ut en avskjedsmelding \n');
end
  
```

Dette kalles funksjonsstubber og er en teknikk som brukes når man skriver i en topp-ned metode. Stubben muliggjør at man kan skrive funksjonen personProgram som f.eks. kaller på process før process-funksjonen er skrevet ferdig. I senere oppgaver skal fprintf-funksjonen byttes ut med den faktiske implementasjonen av funksjonen.

a)

Lag funksjonen `personProgram()`. Funksjonen skal gjøre følgende:

1. Skrive ut velkomstmelding
2. Skrive ut meny
3. Be brukeren om et tall som representerer et valg i menyen
 - a. Hvis tallet er 0, avslutt
 - b. Ellers: Kall på funksjonen `process`
4. Repter steg 2-4 så lenge brukeren ikke vil avslutte.

Eksempel på utskrift:

```
Skriver ut en velkomstmelding
Skriver ut en meny
Velg et tall: 1
Du har valgt 1
Skriver ut en meny
Velg et tall: 0
Skriver ut en avskjedsmelding
```

b)

Fyll ut funksjonen `printHeader()` slik at den skriver ut en passende velkomstmelding. Hovedmenyen skal se slik ut

```
1. Hent database fra fil
2. Lagre database til fil
3. List alle personer
4. Legg inn ny person
5. Endre en person
0. Avslutt programmet
```

c)

Lag funksjonen `printMenu()` som skriver ut menyen.

d)

Fullfør `printFooter()` slik den skriver ut en passende avskjedsmelding.

e)

Definer funksjonsstubbene `processLoad()`, `processStore(P)`, `processListPersons(P)`, `P = processNewPerson(P)` og `P = processChangePerson(P)`. Legg inn en `fprintf` som forteller hva hver av funksjonene skal gjøre. Feks: `processLoad` har `fprintf('Laster fil');`.

PS: `P` er en vektor hvor hvert element er en personstruktur lik den i [Persondatabase](#).

f)

Skriv om `P = process(P, choice)`-funksjonen slik at den kaller riktig underfunksjon ved gitt valg. Hvis valget er < 1 eller > 5 , skal funksjonen skrive ut 'Ugyldig valg!'.

g)

Fyll ut `P = processLoad()`-funksjonen. Den skal be brukeren om å fylle inn et navn på en fil. Så skal den kalle på `loadfile`-funksjonen fra [Persondatabase](#) med navnet som argument. Funksjonen skal returnere vektoren som `loadfile` returnerer.

h)

Fullfør `processStore(P)`. Den skal be brukeren om å fylle inn et navn på en fil. Så skal den kalle `store`-funksjonen (fra [Persondatabase](#)) med navnet og parameteren `P`.

i)

Skriv ferdig `processListPersons(P)`. Funksjonen skal kalle funksjonen `listPersons` fra [Persondatabase](#) med parameteren `P` som argument.

j)

Fyll ut `P = processNewPerson(P)`. Den skal kalle på `promptPerson` fra [Personstruct](#). Personstrukturen som `promptPerson` returnerer skal legges til på slutten av parameteren `P`. Merk at parameteren `P` skal også returneres.

k)

Fullfør `P = processChangePerson(P)` funksjonen. Den skal først be brukeren om å skrive inn et tall, hvor tallet representerer indeksen i person-listen som personen du vil endre på, ligger. Deretter skal funksjonen skrive ut denne personen. Så kaller den på `promptPerson` for å la brukeren skrive inn oppdatert informasjon. Overskriv til slutt personen i parameteren `P`. Denne funksjonen skal også returnere `P`.

l)

Det er vanlig at et program spør om du vil lagre ulagrede endringer før programmet avsluttes. Implementer dette i programmet.

m)

Legg til et menyvalg for å søke etter en person på navn og implementer funksjonaliteten for dette. Søket skal printe ut personen med `printPerson` funksjonen.