

# Hvordan gjør et moderne OS det mulig å kjøre flere prosesser samtidig på én CPU?

3 


Venter til prosessen som kjører gjør I/O og gir da CPU'en til neste prosess i ready-list

1 

Overlater timesharingen til hardware-algoritmer

1 

Hver prosess gir frivillig fra seg tilgangen til CPU til en annen prosess etter en timeslice

1 

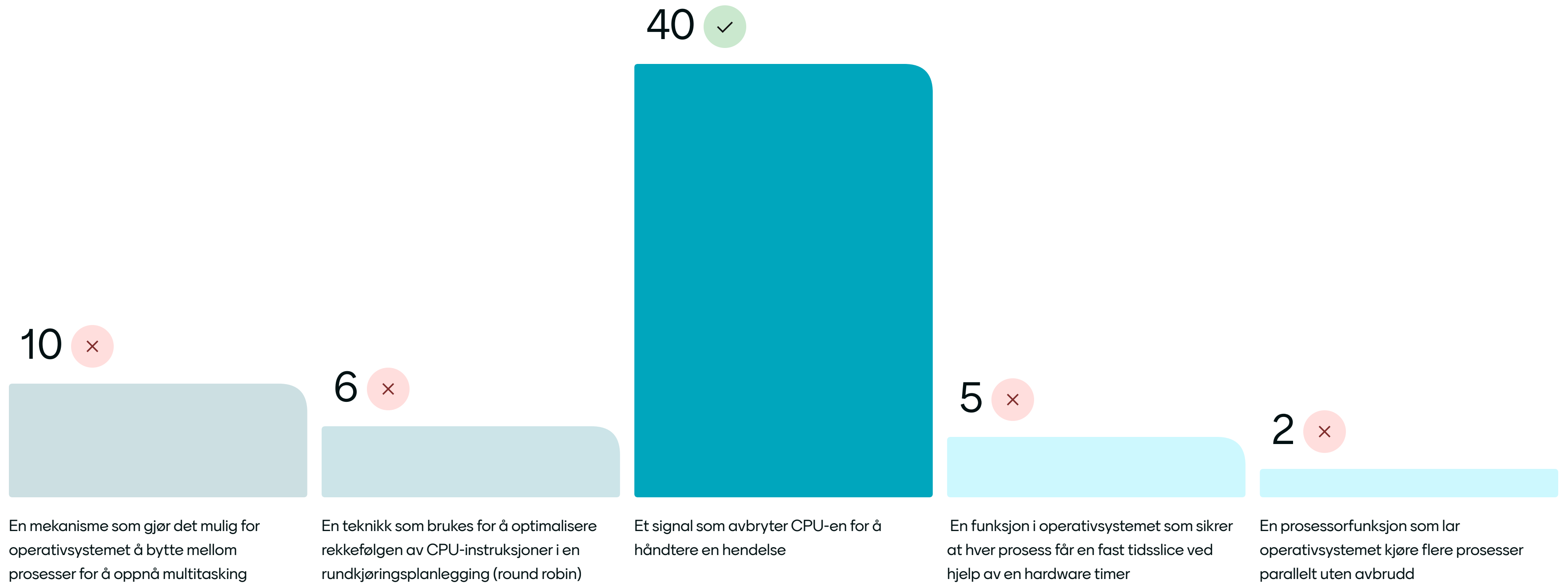
Kjører en while-løkke som plukker prosesser fra ready-list og sender med en tidsparameter til hver prosess som startes

58 

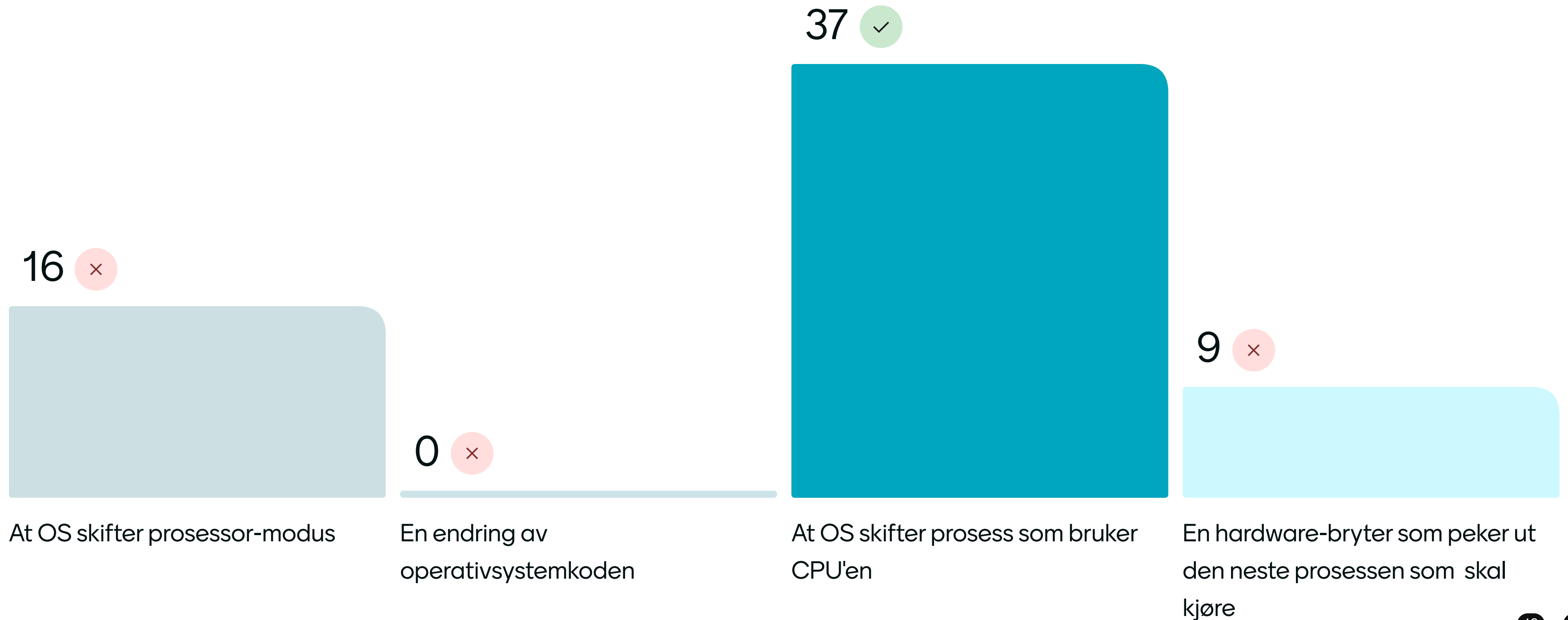
Bruker en hardware timer til å gi begrensede tidsintervall til brukerprosessene



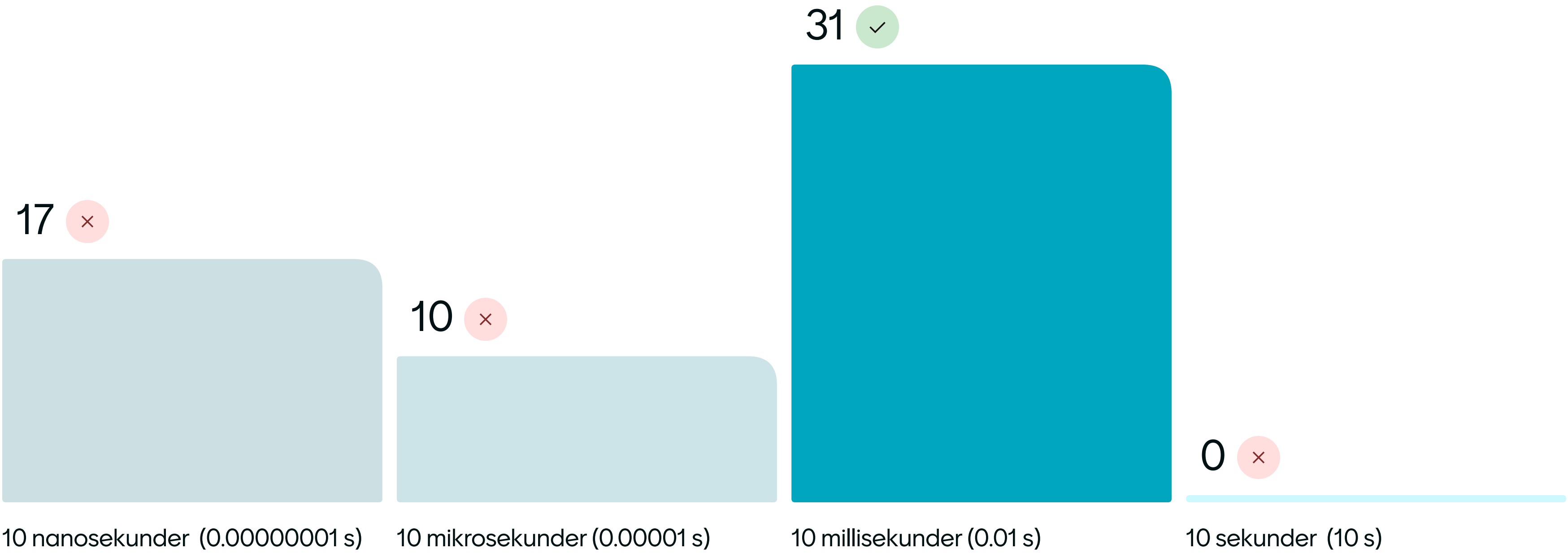
# Hva er et interrupt?



# Hva er en context switch?



Omtrent hvor lang er den minste tidsenheden en prosess kan tildels (tidskvantum) i Linux scheduling?



# Hvorfor kan ikke OS få én vanlig sekvensiell prosess til å gå fortere ved å kjøre den på to CPUer?

2% 

Fordi et OS kun styrer en CPU av gangen. Flere CPUer krever flere OS.

2% 

Fordi OS ikke kan flytte en prosess fra en CPU til en annen etter at prosessen har startet.

4% 

Fordi OS ikke kan hente ut instruksjoner fra samme sted i RAM og kjøre det på to CPUer.

0% 

Fordi en deadlock mellom CPUene da lett kan oppstå.

0% 

Fordi to prosesser ikke kan ha samme navn.

92% 

Fordi OS ikke kjenner den indre logikken i prosessen og må la instruksjonene kjøres sekvensielt.