

9.4 Summary

Monday, April 27, 2020 6:35 PM

OS-et har tre former for tidsstyring.

- Langsiktig tidsstyring avgjør når nye prosesser legges til systemet.
- Mediumsiktig tidsstyring er en del av swapping-funksjonen og avgjør når et program skal enten legges delvis, eller fullstendig i hovedminnet så det kan kjøres.
- Kortsiktig tidsstyring avgjør hvilke prosesser i Ready-tilstand som skal kjøres neste gang på prosessoren.

Tidsstyringsalgoritmer må avgjøre i hvilken rekkefølge ulike prosesser/tråder som ønsker adgang til en felles ressurs, skal få adgang til en gitt ressurs – som regel prosessorer.

Det er flere algoritmer som har blitt utviklet for kortsiktig tidsstyring:

- **First-Come-First-Served:** Velger prosessen som har ventest lengst.
- **Round Robin:** Bruker *time-slicing* for å begrense kjørende prosesser til korte utbrudd av prosessortid, og rotere mellom prosesser i Ready-tilstand.
- **Shortest Process Next:** Velger prosessen med kortest forventet prosesseringstid, og har ikke avbrudd.
- **Shortest Remaining Time:** Velger prosessen med kortest gjenværende prosesseringstid. En prosess kan bli avbrutt når en annen prosess blir klar.
- **Highest Response Ratio Next:** Baserer tidsstyringsvalget på et estimat av normalisert gjennomløpstid.
- **Feedback:** Etabler et sett med tidsstyrings-køer og tildel prosesser til køer basert på execution-historie og andre kriterier.

GENERELT FOR KORTSIKTIG TIDSSTYRINGS ALGORITMER

Ulike situasjoner vil føre til at ulike valg blir naturlige:

- **FCFS:** Kan brukes i satsvise systemer - altså systemer uten interaktiv tilgang for brukere til programmene/applikasjonene som kjøres, hvor høy gjennomstrømning ikke oppfattes som viktig
- **SPN:** Kan brukes i satsvise systemer - altså systemer uten interaktiv tilgang for brukere til programmene/applikasjonene som kjøres, hvor en viss gjennomstrømning ønskes oppnådd gjennom prioritering av små prosesser
- **RR:** Bør brukes i systemer med interaktiv tilgang for brukere til programmene/applikasjonene som kjøres
- **SRT:** Kan brukes i satsvise systemer - altså systemer uten interaktiv tilgang for brukere til programmene/applikasjonene som kjøres, hvor en viss gjennomstrømning ønskes oppnådd gjennom prioritering av både små og store prosesser