

Velkommen til
INF2270
Datamaskinarkitektur

Motto:

Hvordan er en datamaskin bygget opp?

Forelesere

Philipp Häfliger (hafliger@ifi.uio.no) fra gruppen for
Nanoelektronikk.

Dag Langmyhr (dag@ifi.uio.no) fra forskningsgruppen
Objektorientering, modellering og språk.

Første time

- Introduksjon til kurset
 - Hva handler kurset om?
 - Pensum
 - Kursopplegg

I pausen

- Registrering av oppmøte

Andre time

Dag Programmeringsspråket C (seminarrom Perl)

Philipp Logiske kretser (*her* i Simula)

Forelesningene

Siden dere har ulik bakgrunn, blir opplegget:

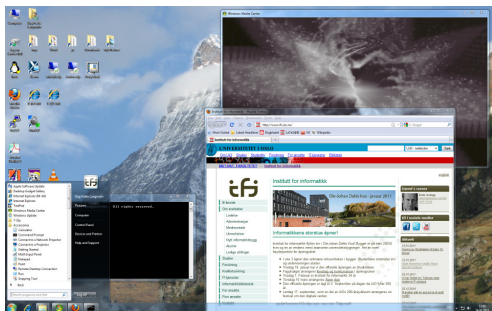
Uke 3½-6½ undervises parallelt i C-programmering (i *Perl*)
og kretsteknikk (i *Simula*)

Resten av kurset undervises felles.

Hva skjer inni en datamaskin?

Grunntanken med kurset

Sett fra brukerens side utfører datamaskinen mange avanserte programmer, men dypt inne i maskinen er alt som skjer at strøm i diverse transistorer slås av og på. Hva er *sammenhengen*?



Datamaskinens nivåer

Datamaskinen er konstruert i ulike nivåer som bygger på hverandre:

I kurset vil vi presentere de ulike nivåene og sammenhengen mellom dem.

Høynivåspråk	Nivå 5
Assemblerspråk	Nivå 4
Operativsystem	Nivå 3
Maskininstruksjoner	Nivå 2
Mikroarkitektur	Nivå 1
Digital logikk	Nivå 0

Lærebøker

Læreboken er

Computer Systems — a programmer's perspective (second edition) av Randal E Bryant og David R O'Hallaron; Pearson 2011; ISBN 0-13-713336-7.

I tillegg har foreleserne skrevet et kompendium. Det ligger på kursets nettside men gjennom semesteret vil kompendiet bli oppdatert og utvidet og etter hvert delt ut til dere.

Hjemmeside

Kursets hjemmeside

<http://www.uio.no/studier/emner/matnat/ifi/INF2270/v11/>
er hovedkanalen for løpende informasjon om kurset.

- Plansjer fra forelesningene blir lagt på hjemmesiden.
- Ukeoppgaver legges ut hver uke; senere kommer også løsningsforslag.
- Kompendiet finnes her.
- Viktige beskjeder dukker opp ved behov.

Gruppeundervisning

Hver uke blir det presentert diverse oppgaver for å underbygge det foreleste stoffet. Ukeoppgavene er også pensum! (Og så er de til god hjelp når man skal løse de obligatoriske oppgavene.)

Tanken bak gruppene er at

- dere skriver ut oppgavene før samlingen og tar dem med,
- dere løser ikke oppgavene på forhånd,
- på de ukentlige samlingene deles dere inn i smågrupper på 3-5 personer som løser oppgavene i fellesskap.

Obligatoriske aktiviteter

Svært lite av kurset er obligatorisk, kun dette:

- Det er obligatorisk fremmøte på første forelesning (altså i dag). Registrering skjer her i salen i pausen.
- Det er to obligatoriske oppgaver som skal løses til fastsatte frister; følg med på hjemmesiden.
- Eksamen!

Programvare

Kretskonstruksjon benytter et eget program — dette vil bli kunngjort senere.

C- og assemblerprogrammering bruker GNUs gcc som finnes på alle Linux- og Mac-maskiner.

På Ifi-DVDen <http://www.ifi.uio.no/ifi-dvd/> finnes gcc som en del av CygWin som kan kjøres på en Windows-maskin hjemme.

Hvilke forventninger har dere?

Forventninger

Hva kan dere forvente å få ut av å ta kurset?

- Kunnskap om hvorledes en datamaskin er bygget opp og fungerer.

Dette vil gjøre dere til bedre programmerere og brukere av datamaskiner.

- Lære å programmere Intels prosessorer.
- Lære å konstruere egne kretser.

Hvilke forventninger har dere?

Et eksempel

Denne løkken bruker 1,29 s når SIZE er 16 000:

```
for (i1 = 0; i1 < SIZE; ++i1)
  for (i2 = 0; i2 < SIZE; ++i2)
    a[i1][i2] += b[i1][i2];
```

mens denne bruker 9,76 s:

```
for (i2 = 0; i2 < SIZE; ++i2)
  for (i1 = 0; i1 < SIZE; ++i1)
    a[i1][i2] += b[i1][i2];
```

Hvorfor?



Hva forventer vi av dere?

Programmeringsferdigheter

Kunne skrive enkle programmer i Java eller et annet programmeringsspråk

Annen bakgrunn

Enten

- kunne programmere i C (f eks fra INF1060) eller
- kjenne til logiske kretser (f eks fra INF1400)