Objektorienterad Programmering (TDDC77)

Föreläsning I: kursinfo, att programmera datorer, första programmet

Ahmed Rezine

IDA, Linköpings Universitet

Höstermin 2016

Outline

Hemsida

Organization

Examination

Webreg

Programmering

Kursinfo

- hemsida: `http://www.ida.liu.se/~TDDC77`
- imperativ och objektorienterad programmering
- föreläsningar, labbar, projekt, datorbaserat tenta, basgruppsarbete
- vi kommer att använda Java genom hela höst terminen

Outline

Hemsida

Organization

Examination

Webreg

Programmering

Kursinfo: organization

`https://www.ida.liu.se/~TDDC77/material/tidsplan.ht16.pdf`

- 13 föreläsningar över HT1 och HT2: helt frivilliga, kommer att följa (ungefär) samma övergripande upplägg som de från 2015, fast med flera ändringar.
- cirka 30 laborationstillfällen för 7 obligatoriska labbar:
  - själva tillfällen är ej obligatoriska, förutom för att demonstrera laborationsuppgifter och projektet
  - fråga om hjälp, diskutera laborationer och projektuppgiftet

Outline

Hemsida

Organization

Examination

Webreg

Programmering

Kursinfo: examination

Efter att ha klarat Stone:

1. Sju laborationsuppgifter (HT1 och HT2). 3hp. LAB (U, G)
2. Ett programmeringsprojekt (HT2). 2hp. UPG (U,3,4,5)
3. En datorbaserat tenta. 1hp. DAT (U,3,4,5)
4. Basgruppsarbete kring vinjetter. 2hp. BAS (U,G)

- Ett godkänt på alla moment krävs för att få ett slutbetyg
- I så fall, bestäms slytbetygen av UPG och DAT
- Examination är alltid individuellt,
- Arbetet i LAB och UPG sker i grupper av två personer.

Examination: regler

- Ni får inte lämna in kopierat (även modifierat) lösningar
- Varje student måste kunna visa att hon/han förstår alla detaljer i lösningen
- Om någon form av fusk misstänks, är läraren skyldig att göra en anmälan till universitetets disciplinnämnd
- Konsekvenserna kan bli varning eller avstängning från studierna

Outline

Hemsida

Organization

Examination

Webreg

Programmering

Kursinfo: webreg

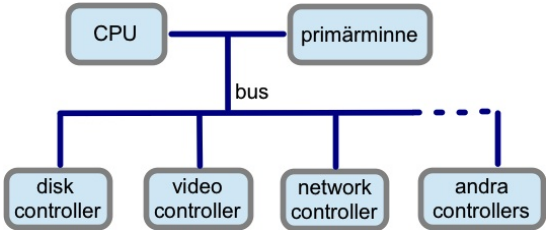
- "Webreg" rapportering av LAB, UPG och DAT.
  1. LAB: i grupper av två personer innan 2016-09-05, `https://www.ida.liu.se/webreg3/TDDC77-2016-1/LAB`
  2. UPG: i grupper av två personer innan 2016-11-07, `https://www.ida.liu.se/webreg3/TDDC77-2016-1/UPG`

Kursinfo: Problem?

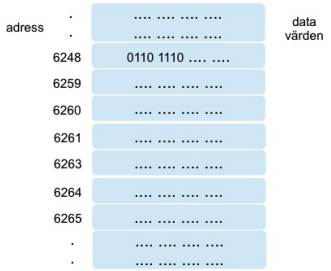
- Frågor som rör det vi lär oss i kursen kan ställas till mig eller till laborationsassistenterna
- Övriga frågor, problem, administrativa saker osv kan man diskutera med mig
- Vill man diskutera mig så kan man prata med Ola Leifler `ola.leifler@liu.se` som är terminsansvarig.

Datorer överallt ...

En “vanlig” dator ...

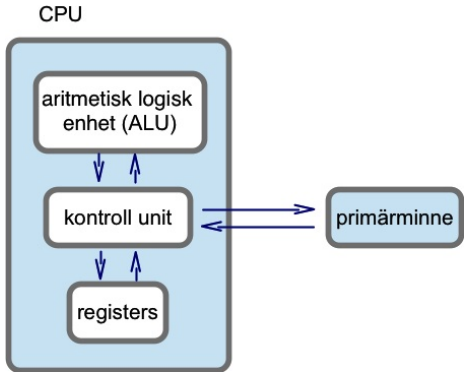


Minnet ...

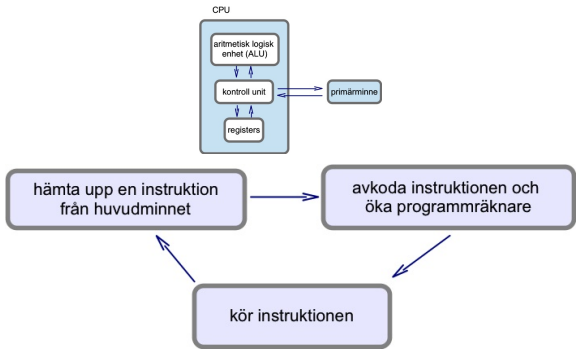


byte	B	$2^0 = 1$
kilobyte	KB	$2^{10} = 1024$
megabyte	MB	$2^{20} = 1,048,576$
gigabyte	GB	$2^{30} = 1,073,741,824$
terabyte	TB	$2^{40} = 1,099,511,627,776$
petabyte	PB	$2^{50} = 1,125,899,906,842,624$

CPU ...



Hämta-avkoda-kör cykeln



Outline

Hemsida

Organization

Examination

Webreg

Programmering

Logga in ..

LiU kontot:

- de som har aktiverat deras LiU konto: bra jobbat! använd den...
- de andra: vänligen gör det så snabbt som möjligt. Använd de tillfälliga konton. OBS: alla filer raderas varje gång ni logga in.

Logga in och skapa en katalog

1. under windows:
   1.1 ladda ner "putty" från "`http://www.chiark.greenend.org.uk/~sgtatham/putty`"
   1.2 öppna "putty" och koppla upp dig med "ssh" och "username@remote-und.ida.liu.se"
2. under MacOS eller Linux:
   2.1 öppna en terminal
   2.2 skriv "ssh username@remote-und.ida.liu.se" och logga in.
3. nu har du tillgång till ditt hemkatalog på IDA.
4. skappa en katalog, säg "mina_tddc77_program" med "mkdir mina_tddc77_program"
5. kom in i katalogen med "cd mina_tddc77_program"

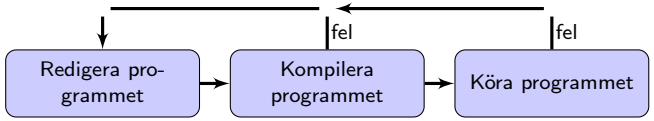
Första programmet!

- i terminalen:
- skapa/öppna Hej.java i emacs med **"emacs Hej.java"**
- spara filen med: **"C-x C-s"**
- klippa/kopiera;klistra? **"C-w"** / **"M-w"**; **"C-y"**
- avsluta? **"C-x C-c"**

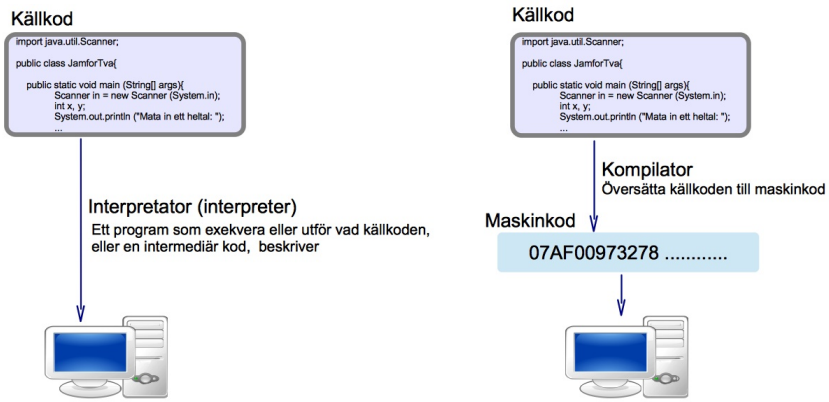
```
/* Programmet visa
   hur man skriva ut
 */
class Hej {

    // Skriv ut ett meddelande

    public static void main(String[] args){
     System.out.println("Hej på er TDDC77:are!!!!!");
    }

}
```

Kompilera och köra programmet

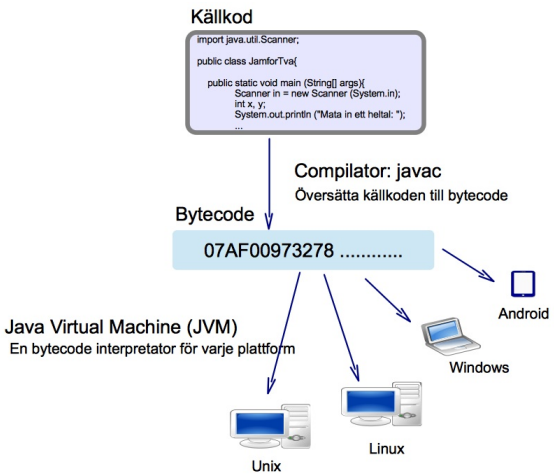
- skapa Hej.java och skriv programmet
- kompilera med **"javac Hej.java"**
- rätta fel och repetera tills du lyckas kompilera ditt program
- kör med **"java Hej"**
- syntax och semantiken



"Interpretera" eller "kompilera" källkod

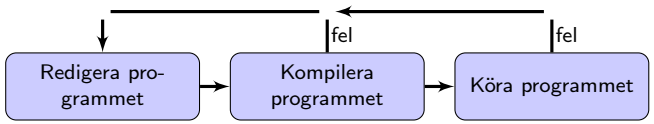


Kompilatorer och JVMer



Kompilera och köra programmet

- skapa Hej.java och skriv programmet
- kompilera med **"javac Hej.java"**
- rätta fel och repetera tills du lyckas kompilera ditt program
- kör med **"java Hej"**
- syntax och semantiken



Ge exakta instruktioner till datorn

- Exempel: hur många ord finns det i "programmering är kul" ? hur gjorde ni?
- Hur många ord i " cd-skivan , som ligger i laptop-väskan,är trasig !" ?
- Datorer måste få exakta "instruktioner" !!

Det är viktigt att förstå att en dator gör precis vad den är sagt att göra, och INGENTING ANNAT ! Därför, måste datorer få exakta "instruktioner" !!

Exakta instruktioner

- Analysera din uppgift och försök att hitta en lösning i ett språk du kan, t.ex. svenska !
- Översätt din lösning till programkod, t.ex. Java
- Försök inte att lösa allt på en gång
- Glöm inte att:
  - Använda papper och penna
  - Prova
  - Ha kul !

Att programera är att bryta ner en uppgift i små steg som en dator kan göra

- Låt användaren skriva in två heltal x och y. Vi vill räkna upp skillnadens absoluta värde (dvs, $|x - y|$) och skriv ut resultatet.
  - Fråga användaren efter ett heltal
  - Läs in ett heltalet x
  - Fråga användaren efter ett heltal
  - Läs in ett heltalet y
  - Om x är storare än y, räkna upp $(x - y)$
  - Om y är storre än x, räkna upp $(y - x)$
  - Skriv ut resultatet

Glömm inte att:

- Använda papper och penna
- Prova
- Ha kul !