

Den fullständiga studiehandledningen för DT001G Informationsteknologi grundkurs

Daniel Bosk*

studyguide.tex 1950 2014-09-03 08:52:07Z danbos

Innehåll

1	Mål	1
2	Kursupplägg	2
2.1	Schema	3
2.2	Introduktionsföreläsning	4
2.3	Laboration L0 Installation	4
2.4	Föreläsning om datarepresentation	4
2.5	Föreläsning om datorarkitektur	4
2.6	Föreläsning om den UNIX-lik terminalen	5
2.7	Laboration L1 Terminalen	5
2.8	Föreläsningar om programmering med Python	5
2.9	Laboration L2 Programmering med Python	7
2.10	Föreläsning om L ^A T _E X	7
2.11	Föreläsning om rapportskrivning och presentationsteknik	7
2.12	Laboration L3 Datorn	7
2.13	Laboration L4 Presentationsteknik	7
2.14	Tentamen	7
3	Examination	7
4	Vad händer om jag ej blir klar i tid?	8

1 Mål

Kursen utgör en introduktion till datateknik och är förberedande för praktiskt inriktade datatekniska utbildningar. Centrala delar är datorns konstruktion, datorkomponenter, grundläggande begrepp och terminologier samt mjukvara. Kursen fokuserar på praktisk användning av datorer och programvaror. Kursen ger även en introduktion till rapportskrivande och presentationer.

Mer specifikt, efter genomgången kurs ska du kunna:

*Detta verk är tillgängliggjort under licensen Creative Commons Erkännande-DelaLika 2.5 Sverige (CC BY-SA 2.5 SE). För att se en sammanfattning och kopia av licenstexten besök URL <http://creativecommons.org/licenses/by-sa/2.5/se/>.

- installera program i UNIX-lik miljö.
- installera program i Windowsmiljö.
- använda programvaror i en UNIX-lik miljö.
- skriva enklare skript.
- redogöra för datorsystems uppbyggnad och funktion.
- beskriva hur information representeras och lagras digitalt.
- genomföra en enklare objektiv undersökning och dra en slutsats av resultatet.
- med akademisk svenska eller engelska skriva en rapport och hålla en muntlig presentation av densamma.

2 Kursupplägg

Den bok som används som huvudlitteratur på kursen är Brookshears bok *Computer Science: An Overview* [3]. Därutöver tillkommer litteratur för en del av laborationerna, denna framgår av läsanvisningarna nedan och i respektive laboration.

Brookshear [3] tar upp grunden inom datateknik, den är värd att läsa i sin helhet – detta rekommenderas! De kapitel och avsnitt som behandlas i denna kurs är dock enbart kapitlen 0–5 i sina helheter, avsnitten 6.1–6.4, samt avsnitten 9.1 och 9.2.

Det går att använda tidigare upplagor av boken, kapitlen som behandlas i kursen utgår från den senaste upplagan [3] och de behandlar följande områden:

- Kapitel 0: introduktion och historia,
- kapitel 1: datalagring och -representation,
- kapitel 2: datamanipulering, datorarkitektur, programexekvering,
- kapitel 3: operativsystem,
- kapitel 5: algoritmer,
- avsnitten 6.1–6.4: programspråk och programmering, samt
- avsnitten 9.1 och 9.2: databaser.

Två kapitel från Silberschatz bok *Operating System Concepts* [8, 9] kommer också att användas. Denna bok kommer att läsas i fulla i nästa kurs, som behandlar operativsystemsteori. Denna bok är valfri att köpa nu, men det rekommenderas då ni ändå måste köpa den senare.

Värt att veta är att det finns två versioner, där den ena har tillägget ”International Student Edition” i titeln [8]. Denna bok saknar kapiteln om distribuerade system och virtualisering. Dessa kapitel behandlas inte i någon kurs, men då de kan vara intressanta för er vill jag göra er uppmärksamma på detta.

Kursvecka	Arbete
1	Kursstart/Föreläsning introduktion L0 Installation
2	Föreläsning om datarepresentation Föreläsning om datorarkitektur
3	Föreläsning om den UNIX-lik terminalen Övning: den UNIX-lik terminalen L1 Terminalen Handledning
4	Föreläsning om programmering med Python, del 1 Övning: Python Handledning
5	Föreläsning om programmering med Python, del 2 Övning: Python L2 Programmering med Python Handledning
6	Handledning
7	Föreläsning om LaTeX Övning: LaTeX Föreläsning om rapportskrivning och presentationsteknik L3 Datorn L4 Presentationsteknik Handledning
8	Handledning
9	Handledning Presentation L4
10	Tentamen

Tabell 1: En sammanställning av kursens moment och när de kommer att genomföras. Tiden är anpassad efter studietakt om halvfart.

Litteraturläsningen kompletteras av ett antal föreläsningar och övningar som ges under kursens gång. Dessa kommer inte att vara heltäckande och för att förstå dem måste du läsa litteraturen enligt anvisningarna nedan.

Kursens lärandemål kommer att examineras med ett antal laborationer och avslutas med en salstentamen.

2.1 Schema

Du finner en sammanställning av kursens schema i tabell 1. Det är naturligtvis valfritt att följa detta schema sånär som på slutdatum för kursens uppgifter och när föreläsningarna ges. Läsanvisningar för respektive moment följer i kommande avsnitt. Undervisningen förutsätter att du följer dessa riktlinjer. Det rekommenderas att du läser igenom materialet innan de lärarledda tillfällena, så att du kan ta upp dina frågor med läraren.

2.2 Introduktionsföreläsning

Föreläsningen går igenom kursstruktur och organisation. Den ger en översikt över undervisning och examination. Den motsvarar således att läsa igenom allt kursmaterial och lite därtill.

Utöver detta täcks även kapitel 0 i *Computer Science: An Overview* [3]. Kapitlet introducerar ämnena datateknik och datalogi (datavetenskap). Det ger även en historisk överblick av området vilket är bra för att förstå varför området är som det är och dess framtida utveckling.

2.3 Laboration L0 Installation

För att genomföra denna laboration bör du ha läst kapitlet om operativsystem och bootprocessen [3, kapitel 3].

Innan du genomför laborationen bör du också läsa igenom dokumentationen för installationen av Ubuntu [11], detta är inte för att installationen är svår att genomföra utan för att du ska kunna fundera igenom dina beslut på förhand.

När Ubuntu väl är installerat finns dokumentationen [12] som stöd för att börja använda systemet. Det kan vara bra att orientera sig i denna för att senare enkelt hitta vid behov. Det rekommenderas att läsa igenom de första fem avsnitten – ”Welcome to Ubuntu” till och med ”Log out, power off, switch users” – innan installationen. Du ska även läsa om hur program installeras [12, se Install additional software].

Om du använder Windows och vill testa att installera programmen även där finns en instruktion för att installera L^AT_EX för Windows [2]. Hur L^AT_EX installeras under Ubuntu täcks senare i lydelsen.

2.4 Föreläsning om datarepresentation

Föreläsningen tar upp kapitel 1 ”Data Storage” i [3], som handlar om fysisk lagring av data på olika media.

Innehållet kommer även att kompletteras med delar från kapitlen 2 och 6 i [1] som förklarar den logiska representationen av data. Kapitel 2 tar logik och bevis. Då en dator är strikt baserad på logiska operationer är detta en viktig grund. Bevis är helt enkelt tillämpning av de logiska reglerna och detta behövs för att förstå bevisen i kapitel 6.

Kapitel 6 täcker talsystem och förklarar varför vi kan använda datorns logiska system för att räkna med godtyckliga tal. Logiska operationer samt representation och beräkningar av tal är vad en dator gör när program körs.

2.5 Föreläsning om datorarkitektur

Föreläsningen behandlar innehållet i kapitlen 2 och 3 i *Computer Science: An Overview* [3] samt kapitel 1 och 2 i *Operating System Concepts* [8, 9]. Innehållet behandlar datamanipulering, datorarkitektur, programexekvering och operativsystem.

Datamanipulering är en fortsättning av datarepresentationen. Här behandlas de operationer som är möjliga att utföra på datat och hur de är konstruerade genom hårdvaran.

Datorarkitektur behandlar systemets struktur, hur de olika hårdvarudelarna i ett datorsystem förhåller sig till varandra. I samband med detta behandlas

även hur hårdvaran kan exekvera program samt huvudprogrammet som körs i ett datorsystem – operativsystemet.

2.6 Föreläsning om den UNIX-lik terminalen

Föreläsningen behandlar den UNIX-lik terminalen. Som framgått i datateknikens historia har terminalen haft en betydande roll, och det är faktiskt så att det fortfarande är få gränssnitt som överträffar terminalens effektivitet.

Den litteratur som täcker detta är kapitel 1, 12.1, 12.5-12.7 och 12.10 i *UNIX and Linux system administration handbook* [6] och självklart tillhörande manualsidor.

2.7 Laboration L1 Terminalen

För att genomföra denna laboration bör du ha läst kapitlet om operativsystem [3, kapitel 3]. Det rekommenderas att du även läser kapitel 1, 12.1, 12.5–12.7 och 12.10 i *UNIX and Linux system administration handbook* [6]. Den senare boken ingår egentligen inte i kurslitteraturen för denna kurs, men då boken ingår i kommande kurser kan ni med fördel nyttja den redan nu.

För dokumentation om olika program i UNIX-lik system används kommandot `man`. Det tar namnet på att annat kommando som du vill ha dokumentation för som argument. Om vi till exempel vill ha dokumentation om just `man` självt skriver vi `man man` och får resultatet i listning 1 på nästa sida. Manualsidorna är indelade i sektioner, denna ges som en siffra inom parentes efter namnet på manualsidan – exempelvis `man(1)`. För att specificera en särskild sektion anges sektionen innan namnet på manualsidan (kommandot) som argument till `man`, exempelvis `man 1 man`. Oftast behövs dock inte detta, det är bara när ett uppslagsnamn finns i flera sektioner. Det framgår i listning 1 på nästa sida att det är `man(1)` som ges av `man man`, alltså samma resultat som vid `man 1 man`. Dessa manualer finns även tillgängliga online på URL

<https://www.kernel.org/doc/man-pages/>.

De manualsidor som bör läsas översiktligt i förväg är `bash(1)` och `man(1)`.

För dokumentation om kommandona i Windows kan följande sida användas:

<http://technet.microsoft.com/en-us/library/bb490890.aspx>

2.8 Föreläsningar om programmering med Python

Programmering och algoritmer är en central del inom datateknik och datalogi. Dels behövs programmering för att kunna använda ett datorsystem från första början. Men även när det finns program som går att använda utan programmeringskunskaper, kommer programmering att underlätta många uppgifter avsevärt.

Studiet av algoritmer är också centralt inom området. Algoritmer implementeras i kod med programmering, så en algoritm handlar om hur någonting kan utföras systematiskt och (gärna) så effektivt som möjligt. Dessa rör alltifrån sortering av namn till routingalgoritmer som ger effektiva nätverk.

Föreläsningarna behandlar kapitel 5, 6 och 8 i *Computer Science: An Overview* [3]. Utöver detta tillkommer även avsnitten 2–6 i *Pythonkramaren del 1: Programmering för teknologer* [4].

```

1 /home/danbos$ man man
2 MAN(1)                Manual pager utils                MAN(1)
3
4 NAME
5     man - an interface to the on-line reference manuals
6
7 SYNOPSIS
8     man [-C file] [-d] [-D] [--warnings[=warnings]]
9         [-R encoding] [-L
10        locale] [-m system[,...]] [-M path] [-S list] [-e
11        extension] [-i|-I]
12        [--regex|--wildcard] [--names-only] [-a] [-u]
13        [--no-subpages] [-P
14        pager] [-r prompt] [-7] [-E encoding]
15        [--no-hyphenation] [--no-justifi-
16        cation] [-p string] [-t] [-T[device]]
17        [-H[browser]] [-X[dpi]] [-Z]
18        [[section] page ...] ...
19     man -k [apropos options] regexp ...
20     man -K [-w|-W] [-S list] [-i|-I] [--regex] [section]
21         term ...
22     man -f [whatis options] page ...
23     man -l [-C file] [-d] [-D] [--warnings[=warnings]]
24         [-R encoding] [-L
25         locale] [-P pager] [-r prompt] [-7] [-E encoding]
26         [-p string] [-t]
27         [-T[device]] [-H[browser]] [-X[dpi]] [-Z] file ...
28     man -w|-W [-C file] [-d] [-D] page ...
29     man -c [-C file] [-d] [-D] page ...
30     man [-hV]
31
32 DESCRIPTION
33     man is the system's manual pager. Each page argument
34     given to man is
35     normally the name of a program, utility or function.
36     The manual page
37     associated with each of these arguments is then found
38     and displayed. A
39     section, if provided, will direct man to look only
40     in that section of
41     the manual. The default action is to search in all
42     of the available
43     sections, following a pre-defined order and to show
44     only the first page
45     found, even if page exists in several sections.
46
47     The table below shows the section numbers of the
48     manual followed by the
49     types of pages they contain.
50     [...]
51 /home/danbos$

```

Listning 1: Listningen av resultatet vid körningen av kommandot man man.

2.9 Laboration L2 Programmering med Python

Innan du påbörjar laborationen ska du ha läst kapitel 1, 2, 5 och 6.1–6.4 i [3] samt avsnitt 2–6 i *Pythonkramaren del 1: Programmering för teknologer* [4]. Du bör även ha gjort flertalet övningar i litteraturen.

2.10 Föreläsning om L^AT_EX

Föreläsningen introducerar grundläggande LaTeX, det som behövs för akademisk rapportskrivning. LaTeX är ett system för typsättning av dokument, det vill säga att dokumentet framställs på ett sätt som drar nytta av kunskaperna från århundraden av boktryckning – hur texten blir så behaglig och effektiv som möjligt att läsa. (Detta till skillnad från vanliga ordbehandlare, se *A Few Notes on Book Design* [13] för en diskussion om detta.) Andra fördelar är att det är enkelt att systematisera och fokus hamnar enbart på innehållet, LaTeX sköter hur innehållet presenteras, numrera korrekt, hantera referenser, etc.

Inför föreläsningen bör du ha läst kapitlen 1–3 i *The Not So Short Introduction to L^AT_EX 2_ε: or L^AT_EX 2_ε in 157 minutes* [7].

2.11 Föreläsning om rapportskrivning och presentations-teknik

Föreläsningen behandlar grundläggande rapportskrivning, motsvarande innehållet i *Rapportmall för tekniska rapporter* [5], och tar dessutom upp presentationsteknik. Materialet från kapitel 3, 4 och 5 i *The BEAMER class: User Guide for version 3.24* [10] berörs också.

2.12 Laboration L3 Datorn

Du ska först ha läst igenom kapitlen 0–4 i [3]. Därefter ska du ha läst igenom kapitel 1–3 i *The Not So Short Introduction to L^AT_EX 2_ε: or L^AT_EX 2_ε in 157 minutes* [7].

2.13 Laboration L4 Presentationsteknik

För denna uppgift ska du läsa igenom kapitel 5 ”Guidelines for Creating Presentations” i *The BEAMER class: User Guide for version 3.24* [10]. Det rekommenderas dock att du läst igenom även kapitlen 3 och 4.

2.14 Tentamen

Tentamen i denna kurs examinerar den teori som behandlats, det vill säga samtlig litteratur som behandlats under kursens gång.

3 Examination

Kursen examineras med med inlämningsuppgifter (laborationer), se lydelserna för detaljer, och en skriftlig salstentamen.

Den första uppgiften L0 rapporteras in som moment I010 i ladok. Övriga laborationer L{1,2,3} rapporteras in gemensamt som moment I110 i ladok, det

vill säga samtliga måste vara godkända innan detta kan ske, och detta ger 3.5 högskolepoäng. Alla dessa uppgifter betygsätts godkänt eller underkänt.

Den muntliga presentationen rapporteras in som P110, detta ger en (1) högskolepoäng, och betygsätts godkänt eller underkänt.

Den avslutande salstentamen betygsätts med hela betygsckalan, A-E för godkänt och F eller Fx för underkänt. Tentamen rapporteras som moment T110 i ladok och ger tre (3) högskolepoäng.

4 Vad händer om jag ej blir klar i tid?

Slutdatumena på denna kurs är av yttersta vikt. Du måste ha genomfört introduktionsuppgiften L0 inom dess givna slutdatum, om du inte gör detta kommer du att avregistreras från kursen och din plats kommer att ställas till förfogande för andra sökande.

Vad gäller den övriga examinationen på kursen kommer det för redovisningar att ges ett presentationstillfälle under kursens gång. Därefter ges ytterligare två presentationstillfällen, dessa förläggs inom ett år. Alla dessa tillfällen kommer att finnas i kursens schema (i studentportalen).

De slutdatum som finns för dessa tillfällen är strikta. Om du missar slutdatumet för ett tillfälle hänvisas du till nästa redovisningstillfälle. Efter det tredje redovisningstillfället hänvisas till redovisningstillfällena under nästkommande kursomgång.

För skriftliga inlämningsuppgifter gäller att dessa rättas en gång under kursens gång, senast i samband med slutdatum för inlämning, därefter ytterligare två gånger i de kommande omtentamensperioderna. Totalt erbjuds tre försök per år. Därefter hänvisas till nästa kursomgång.

För skriftlig salstentamen gäller att du måste anmäla dig i förväg. Du kan ansöka om att skriva tentamen på annan ort, detta måste du dock ansöka om i god tid och du måste ordna plats själv. Tentamen skrivs endast under de tider som finns i schemat, inga undantag kan göras. Se instruktionerna i studentportalen för vidare detaljer.

Ingen handledning planeras efter kursens slut, det vill säga efter det sista schemalagda handledningstillfället. Om du inte hinner bli klar med uppgifterna inom kursens tidsramar och du vill vara garanterad handledning av lärare krävs att du omregistrerar dig på nästa kurstillfälle. Omregistrering på kurstillfälle sker i mån om plats, alla förstagångssökande och reserver kommer att prioriteras.

Om du vid kursslut har majoriteten av kursens moment kvar att göra hänvisas du direkt till nästa kursomgång, då krävs omregistrering. Huruvida din prestation är tillräcklig för att enbart komplettera eller om omregistrering krävs bedöms av ansvarig lärare.

Om du känner att du inte kommer att hinna bli klar med kursen är det därför bättre att göra ett tidigt avbrott på kursen och söka om den inför nästa kurstillfälle. Tidigt avbrott kan registreras senast tre veckor från kursstart och då kommer du att räknas som en förstagångssökande nästa gång du söker kursen.

Referenser

- [1] Daniel Bosk. "Matematik 1c". I: *En formalisering av matematiken i svensk gymnasieundervisning*. Trita-MAT. MA, ISSN 1401-2278. Stockholm: Institutionen för matematik, Kungl. Tekniska högskolan, 2011. Kap. Bilaga B. URL: <http://kth.diva-portal.org/smash/get/diva2:509959/ATTACHMENT01.pdf>.
- [2] Daniel Bosk och Lennart Franked. *L^AT_EX Usage Instructions*. 2012. URL: <http://ver.miun.se/dokument/instruktioner/latex/windows.pdf>.
- [3] J. Glenn Brookshear. *Computer Science: An Overview*. 11, internationella. Boston: Pearson Addison-Wesley, 2012. ISBN: 978-0-273-75139-7.
- [4] Henrik Eriksson. *Pythonkramaren del 1: Programmering för teknologer*. Kungliga Tekniska högskolan. URL: <http://www.csc.kth.se/utbildning/kth/kurser/DD1344/grudat10/pythonkramaren1.pdf>.
- [5] Mittuniversitetet. *Rapportmall för tekniska rapporter*. 2012. URL: <http://ver.miun.se/latex/miunthes/thesis/thesis.pdf>.
- [6] Evi Nemeth, Garth Snyder, Trent R. Hein och Ben Whaley. *UNIX and Linux system administration handbook*. 4th ed. Upper Saddle River, NJ: Prentice Hall, 2011. ISBN: 978-0-13-148005-6 (pbk. : alk. paper).
- [7] Tobias Oetiker, Hubert Partl, Irene Hyna och Elisabeth Schlegl. *The Not So Short Introduction to L^AT_EX 2_ε: or L^AT_EX 2_ε in 157 minutes*. 2011. URL: <http://www.ctan.org/tex-archive/info/lshort/english/lshort.pdf>.
- [8] Abraham Silberschatz, Peter Baer Galvin och Greg Gagne. *Operating System Concepts*. 9. utg. International Student Version. Hoboken, N.J.: John Wiley & Sons Inc, 2013.
- [9] Abraham Silberschatz, Peter Baer Galvin och Greg Gagne. *Operating System Concepts*. 9. utg. Hoboken, N.J.: John Wiley & Sons Inc, 2013.
- [10] Till Tantau, Joseph Wright och Vedran Miletić. *The BEAMER class: User Guide for version 3.24*. 2012. URL: <http://mirrors.ctan.org/macros/latex/contrib/beamer/doc/beameruserguide.pdf>.
- [11] Ubuntu Community. *GraphicalInstall*. 2012. URL: <https://help.ubuntu.com/community/GraphicalInstall>.
- [12] Ubuntu Documentation Team. *Ubuntu Desktop Guide*. 2012. URL: <https://help.ubuntu.com/12.04/ubuntu-help/index.html>.
- [13] Peter Wilson. *A Few Notes on Book Design*. 2009. URL: <http://mirrors.ctan.org/info/memdesign/memdesign.pdf>.