



[KTH / CSC / Kurser / DD1352 / adk11 / Schema](#)

## Schema och detaljschema i ADK, hösten 2011

Här finns [schemat för kursen](#).

### Detaljschema

Kursen består av 32 föreläsningar och 12 övningar. Alla föreläsningar efter första veckans tre föreläsningar är entimmesföreläsningar. Följande tabell visar vad som preliminärt kommer att behandlas under föreläsningarna och övningarna. För varje föreläsning anges vilka sidor det behandlas i kursböckerna. GT=Goodrich-Tamassia, KT=Kleinberg-Tardos, Moln=Algorithms and Complexity 2007 som har använts tidigare i kursen, Sup=Supplementet Algorithms and Complexity 2009/2011.

#### Period 1

F1 29 augusti (2 timmar)

[Introduktion till kursen](#). [Effektiv kodning](#) och avlusning. Gästföreläsning av Stefan Nilsson. Se även [länksidan](#).

F2 30 augusti (2 timmar)

Repetition av algoritmanalys, beräkningsmodeller, bitkostnad, enhetskostnad. Repetition av [sortering \(merge sort\)](#). (GT: 3-46, 217-238, 268-270, 686-688, KT/Moln: 29-56, 209-221)

F3 31 augusti (2 timmar)

Datastrukturer: repetition, hashning, praktiska datastrukturer, [trie \(animering\)](#). (GT: 55-130, 141-151, 429-438, KT/Moln: 57-65)

Datastrukturer: latmanshashning, [skipplistor](#). (Sup: 77-83, Moln: 595-602)

Ö1 31 augusti

Algoritmanalys.

F4 5 september

Datastrukturer: bloomfilter. Tillämpning: [rättstavning](#).

F5 7 september

Undre gränser. (GT: 239-240, Sup: 17-29, Moln: 535-547)

F6 8 september

Korrekthetsbevis.

Ö2 8 september

Datastrukturer och undre gränser. **Teoriredovisning för labb 1.**

F7 12 september

Grafer: [djupetförstsökning](#), [breddenförstsökning](#). (GT: 287-334, KT/Moln: 73-107)

F8 14 september

Grafer: minimala spännande träd ([Prim](#) och [Kruskal](#)), kortaste stigar ([Dijkstra](#)). (GT: 287-334, 339-353, 360-375, KT/Moln: 137-157)

F9 15 september

Grafer: [maximala flöden](#). (GT: 381-397, KT/Moln: 337-357, 367-373)

Ö3 16 september

Grafalgoritmer.

**Labb 1** 16 september

Konkordans, redovisning.

F10 19 september

Algoritmkonstruktion: dekomposition. (GT: 263-273, KT/Moln: 221-234, 242-246)

F11 20 september

Algoritmkonstruktion: giriga algoritmer, totalsökning. (GT: 257-262, Sup: 31-48, KT/Moln: 115-177, 183-188, Moln: 549-566)

F12 21 september

Algoritmkonstruktion: dynamisk programmering. (GT: 274-281, KT/Moln: 251-290)

Ö4 22 september

Dekomposition och lådpackning. **Teoriredovisning för labb 2.**

F13 26 september

Algoritmkonstruktion: mer dynamisk programmering. (GT: 354-359, KT/Moln: 290-311)

F14 28 september

Algoritmkonstruktion: geometriska algoritmer, [Grahamscan](#). (GT: 572-586)

F15 29 september

Algoritmkonstruktion: sortering i linjär tid. [Räknesortering](#) (GT: 241-244, Sup: 1-6, Moln: 519-524)

Ö5 29 september

Dynamisk programmering.

- [Labb 2](#) 30 september  
Ordkedjor, redovisning.
- F16 3 oktober  
Algoritmkonstruktion: [textsökning](#). (Sup: 7-16, Moln: 525-534, [Pythonkramaren II: 46-48](#))
- [F17](#) 4 oktober  
Algoritmkonstruktion: polynomberäkningar och FFT. (GT: 488-507, KT/Moln: 234-242)
- [F18](#) 6 oktober  
Probabilistiska algoritmer. (GT: 245-247, KT: 707-724)
- [Ö6](#) 6 oktober  
Algoritmkonstruktion. **Teoriredovisning för labb 3.**  
[Sammanfattning av alla algoritmer hittills i kursen](#)
- Mästarprov 1**, senast måndag 10 oktober klockan 9.15!  
Algoritmer.
- [F19](#) 10 oktober  
Reduktioner. (KT: 451-459, Moln: 375-383)
- [F20](#) 12 oktober  
Introduktion till komplexitet, [motivering](#). (GT: 592, KT: 463-466, Moln: 387-390)
- [F21](#) 13 oktober  
Formella definitioner, [turingmaskiner](#). (GT: 593-599)
- [Ö7](#) 13 oktober  
Probabilistiska algoritmer. Reduktioner.
- Period 2**
- [F22](#) 24 oktober  
Oavgörbarhet. (Sup: 49-73, Moln: 567-590)
- Extra visualiseringsföreläsning 24 oktober kl 11-12  
NP-reduktionsvisualisering med [Alvie \(instruktioner\)](#)  
I Alvie finns reduktioner för delmängdssumma (Subset Sum) och hörntäckning (Vertex Cover) visualiserade och bevisade. Reduktionerna finns implementerade i Java i [Teaching machine](#) (klicka på Table of contents, NP-completeness och Run, vänta sedan några sekunder tills maskinen startas; den fungerar ungefär som en grafisk avlusningsmiljö).  
Alvie och Teaching machine är utvecklade av Pierluigi Crescenzi.  
[Reduktionen av 3-cnfsat till delmängdssumma](#) visualiserad med Alvie och sparad som [Flash](#) och [Quicktime](#).
- [F23](#) 25 oktober  
Cooks sats. (GT: 600-602)
- [Ö8](#) 25 oktober  
Genomgång av [lösning till mästarprov 1](#). Oavgörbarhet.
- [Labb 3](#) 28 oktober  
Flöden och matchningar, sista redovisningstillfälle.
- [F24](#) 1 november  
NP-fullständighetsbevis. (GT: 603-609, KT: 466-495, Moln: 390-419)
- [F25](#) 2 november  
NP-fullständighetsreduktioner. (GT: 610-617, KT: 459-463, 497-505, Moln: 383-387, 421-429)
- [Ö9](#) 2 november  
NP-fullständighetsbevis. **Teoriredovisning för labb 4.**
- [F26](#) 8 november  
Mer NP-fullständighetsreduktioner.  
[Reduktionen av 3-cnfsat till hörntäckning](#) visualiserad med Alvie och sparad som [Flash](#) och [Quicktime](#).
- [F27](#) 9 november  
Approximationsalgoritmer. (GT: 618-626, KT: 599-630, 724-727 Moln: 477-508)
- [Ö10](#) 9 november  
NP-fullständiga problem.
- [Labb 4](#) 11 november  
NP-fullständighetsreduktioner, redovisning.
- [F28](#) 15 november  
Mer approximationsalgoritmer.
- [F29](#) 16 november  
[Heuristiska algoritmer. Simulated annealing](#). (KT: 661-670, Moln: 509-518)
- [Ö11](#) 16 november  
Approximationsalgoritmer.
- Mästarprov 2**, senast 22 november klockan 9.15  
Komplexitet.
- [F30](#) 22 november  
Komplexitetsklasser. (KT: 495-497, 531-547, Moln: 419-421, 455-471)
- F31 23 november  
Föreläsning till förfogande: [Fortsättningskurser](#), [Knuths terminologi](#), komplexitet för parallella beräkningar.
- [F32](#) 29 november  
Repetition. Kursens betygssystem.

[Ö12](#) 29 november

Genomgång av [lösning till mästarpöv 2](#). Komplexitetsklasser och repetition.  
Teoritentia 19 december klockan 9-12 i sal F1

[Extralabb](#) 9 och 11 januari 2012

Heuristik för rollbesättningsproblemet, redovisning av frivillig labb.  
Frivillig munta för högre betyg, 12 och 13 januari 2012

[Handledarschema](#)

[Copyright ©](#)

**Sidansvarig:** Viggo Kann [sviggo@nada.kth.se](mailto:sviggo@nada.kth.se)  
Uppdaterad 2012-08-26



[KTH / CSC / Kurser / DD1352 / adk11 / Examination](#)

## Examination i ADK, hösten 2011

Följande betygsriterier tillämpas i kursens examination. Du måste uppfylla alla kriterier på det betyg du ska få.

### Betygskriterier i DD1352 Algoritmer, datastrukturer och komplexitet

Nedanstående betygsriterier bygger på kursens lärandemål och är utformade efter den nya betygsskalan A, B, C, D, E där A är högst betyg. För den som är registrerad på den gamla kursen 2D1352 används betygsskalan 5, 4, 3 där A=5, C=4, E=3.

mål	E	D	C	B	A
<i>utveckla algoritmer med datastrukturer</i>	för enkla problem givet en konstruktionsmetod	för icke-triviala problem givet ledtråd	för icke-triviala problem	för svårare problem	för svårare problem med den metod som passar bäst
	examineras med labbar (för nivå E), mästarpöv 1 och muntlig tenta				
<i>implementera algoritmer med datastrukturer</i>	efter funktions-specifikation och efter detaljerad algoritmisk specifikation, med hänsyn taget till effektivitet				
	examineras med labbar				
<i>analysera algoritmer med avseende på effektivitet</i>	förklara principerna, analysera enklare algoritmer	analysera rekursiva algoritmer med mästarsatsen	analysera svårare algoritmer		
	examineras med labbar (för nivå E), mästarpöv 1, teoritenta och muntlig tenta				
<i>analysera algoritmer med avseende på korrekthet</i>	förklara principerna, förstå ett givet korrekthetsbevis	genomföra enklare korrekthetsbevis	resonera med invarianter och induktion		
	examineras med mästarpöv och muntlig tenta				
<i>jämföra alternativa algoritmer och datastrukturer med hänsyn till effektivitet och pålitlighet</i>					
examineras med labbar och mästarpöv 1					
<i>definiera begreppen P, NP, NP-fullständighet och oavgörbarhet</i>					
examineras med teoritenta och mästarpöv 2					
<i>jämföra problem med hänsyn till komplexitet med hjälp av reduktioner</i>	förklara principerna, utföra enklare reduktioner mellan givna problem	visa NP-fullständighet givet ledtråd	visa NP-fullständighet och oavgörbarhet	göra konstruktionsreduktioner givet ledtråd	göra konstruktionsreduktioner
	examineras med labb 4 (för nivå E), mästarpöv 2 och muntlig tenta				
<i>förklara hur man kan hantera problem med hög komplexitet</i>	förklara behovet	förklara principerna	konstruera enkla heuristiker och totalsökningsalgoritmer	konstruera och analysera enklare approximationsalgoritmer eller heuristiker	konstruera och analysera approximationsalgoritmer eller heuristiker, eller visa undre gränser för approximation
	examineras med teoritenta (upp till betyg C) och muntlig tenta eller labb 4-extrauppgift (för betyg A+B)				

Kursen har fyra obligatoriska moment i Ladok:

- LAB1, datorlaborationer, 3 hp
- MAS1, mästarpöv 1, 1,5 hp
- MAS2, mästarpöv 2, 1,5 hp
- TEN2, tenta, 3 hp

Nedan finns detaljerad information om dessa moment.

### Labbar

Fyra obligatoriska datorlabbar ingår i kursen. Dessa utgör momentet LAB1. Labbarna ska helst göras i tvåpersonsgrupper. En- och trepersonsgrupper kan godkännas i undantagsfall. Varje labb som redovisas senast det labbtillfälle som finns angivet på labben ger en bonuspoäng på tentan. På varje labb finns dessutom ett antal frivilliga teoriuppgifter. Teoriuppgifterna redovisas på övningstillfällena och ger en bonuspoäng var.

Sammanlagt kan alltså labbarna och teoriuppgifterna ge åtta bonuspoäng på tentan. Bonuspoängen gäller på alla tentor på kursen som går inom ett kalenderår räknat från kursstart.

Det finns schemalagda labbtillfällen från och med andra veckan av kursen och till och med den vecka då labb 4 ska redovisas. Det kommer att finnas handledare tillgängliga på dessa labbpass. Börja att göra labbarna i god tid och fråga handledarna om du får problem. Du kan i princip redovisa alla labbarna vid alla labbtillfällen, men under det sista labbtillfället för varje labb kan bara den labben redovisas.

Det finns ett [labbkvitto](#), där labbassen signerar att du är godkänd på labbar, som du ska ta med vid varje redovisning. Här är länkar till labbarna: [labb 1](#), [labb 2](#), [labb 3](#), [labb 4](#).

### Individuella uppgifter: mästarprov

Två obligatoriska individuella uppgifter, *mästarprov*, kommer att delas ut. Dessa ska lösas *individuell* och redovisas både skriftligt och muntligt. Skriftliga lösningar till dessa uppgifter ska lämnas till någon av lärarna eller lämnas in på [studentexpeditionen](#) senast den tid som anges på uppgiftslydelsen. Den muntliga redovisningen kommer att ske senare samma vecka för någon av assistenterna på en tid som ska bokas i förväg på kurswebbsidan.

Varje mästarprov består av tre uppgifter av olika svårighetsgrad. En rätt löst uppgift ger betyg E på momentet, två rätt lösta uppgifter ger betyg C och alla rätt ger betyg A.

Den som inte godkänts på ett mästarprov får möjlighet att göra ett nytt i slutet av kursen, men kan då bara få betyg E på mästarprovet.

Du kan se dina resultat på redovisade uppgifter i kursen i [Rappsystemet](#).

### Tenta

Ordinarietentan går den 19 december 2011 klockan 9 i sal F1. Nästa tillfälle är i vår vid ordinarietentan för kursen DD2352 Algoritmer och komplexitet för F. Därefter blir det en omtenta i juniperioden.

Tentan är en teoritent (om 20 poäng) utan hjälpmedel. För godkänt krävs minst 14 poäng. 17 poäng ger betyg D och 20 poäng ger betyg C. Betyg B och A delas inte ut på teoritentan. Den som redovisat datorlabbarna i tid och har svarat rätt på teoriuppgifterna på labbarna får 8 poängs bonus på tentan.

Skrivtiden är två timmar. Direkt efter tentan vidtar obligatorisk genomgång av lösningarna till tentan och kamraträttning. Rättningen kontrolleras sedan av lärarna och resultatet kungörs samma vecka. Klagomål på rättning av tentan görs till kursledaren. Den som hamnar under men tillräckligt nära gränsen för godkänt på tentan ges möjlighet att komplettera. Kursledaren avgör gränsen för komplettering liksom hur och när kompletteringsuppgifter ska redovisas.

### Du behöver inte anmäla dig till tentan.

#### Muntlig tenta och slutbetyg

Den som fått godkänt på labbarna, båda mästarproven och teoritentan får godkänt på kursen. Slutbetyget bestäms av betygen på samtliga tre betygsatta moment (mästarproven och teoritentan) eventuellt kompletterat med en muntlig tenta och/eller en extralabb (se tabellen med betygskriterier). Den som har fått minst betyg x på alla tre betygsatta moment är värd (minst) betyg x i slutbetyg. Betyget på teoritentan kan höjas till A eller B om man gör extralabben till labb 4 och redovisar på det särskilda redovisningstillfället i vecka 2 2012.

Den som fått minst betyg C på minst två av momenten och minst betyg E på det tredje har möjlighet att gå upp på en muntlig tenta för att få högre betyg. För att få munta måste man ha fått godkänt på den teoritentan som föregår muntan (gäller även omtentor). Den muntliga tentan kan bokas in (efter att teoritentan är rättad) på tider under vecka 2, 2012. Vid den muntliga tentan kommer läraren att kontrollera att du uppfyller alla betygskriterier för det betyg du aspirerar på. Kursböckerna (men inga kompendier eller anteckningar) är tillåtna hjälpmedel.

För att förtydliga slutbetygsberäkningen ges här några exempel på olika sätt att få betyg:

elev	mästarprov 1	mästarprov 2	teoritentan	extralabb	munta	slutbetyg	kommentar
Filemon	E	F	D	-	-	-	inte godkänd på mästarprov 2
Ebon	E	D	E	-	-	E	kan inte gå upp på muntan
Durian	D	C	D	-	-	D	kan inte gå upp på muntan
Cecil	B	D	C	-	C	C	muntade upp mästarprov 2
Beda	B	A	D	B	-	B	valde att inte munta till A
Asta	C	B	C	A	A	A	muntade upp mästarprov 1 och 2

Det finns också en [beskrivning av examinationen som flödesschema](#) (tack till Erik Fahlén).

### Arbetsituationer

Det är meningen att arbetet med momenten i kursen ska motsvara olika arbetsituationer i arbetslivet.

Labbarna tränar olika typer av programutvecklingsarbete:

- Labb 1 är programmering efter en funktionsspecifikation.
- Labb 2 är omprogrammering av ett existerande program så att det ska fungera likadant fast effektivare.
- Labb 3 är programmering efter en detaljerad algoritmisk specifikation.

I alla labbar finns noggranna beskrivningar av format för indata och utdata. Alla labbar har givna effektivitetskrav och utförs som lagarbete (labbgrupper), precis som i arbetslivets programmeringsprojekt.

Mästarproven tränar expertsituationen, alltså situationen som den som vet mest om något på en arbetsplats ställs inför när han får ett problem: det finns ingen att fråga, så han måste komma fram till svaret med egen tankekraft och genom att läsa litteratur. När problemet är löst ska experten förklara lösningen för chefen, både skriftligt och muntligt.

Tentan liknar tyvärr ingen verklig arbetssituation, men den följs av en kamraträttnings-session som är mycket värdefull ur ett pedagogiskt perspektiv.

## Kurskatalog

Kursen har en katalog på skolans Unixdatorer: `/info/adk11`. På denna katalog finns textfiler, programskelett, program och liknande som har med kursen att göra.

[Copyright ©](#)

Sidansvarig: Viggo Kann [viggo@nada.kth.se](mailto:viggo@nada.kth.se)  
Uppdaterad 2012-12-14