



[KTH / CSC / Kurser / DD2458 / popup13 / Kursanalys](#)

## Kursanalys för DD2458, popup13 (VT 2013)

Författare: Per Austrin

### Kursdata

Data har hämtats från RAPP och baserar sig på resultat inrapporterade 2013-06-24

<b>Kurs</b>	DD2458, Problemlösning och programmering under press, 9hp
<b>Examination</b>	programmeringsuppgifter, problemsessioner och anteckningar
<b>Genomförd</b>	VT 2013
<b>Föreläsningar</b>	24 timmar (12 st)
<b>Kurslitteratur (rekommenderad)</b>	Kompendium med föreläsninganteckningar.
<b>Antal studenter</b>	29 varav en kvinna
<b>Prestationsgrad</b>	93%
<b>Examinationsgrad</b>	90%
<b>Ansvarig/föreläsare</b>	Per Austrin
<b>Övriga lärare</b>	Mikael Goldmann, Lukáš Poláček och Marc Vinyals

### Sammanfattning

Jag tog över kursen efter Mikael Goldmann som lämnat KTH. Jag är på det stora hela nöjd med kursen även om det finns skavanker att åtgärda.

### Antagning till kursen och antal studenter

Det blev relativt sent klart att kursen skulle bli av, men den hade ändå 45 sökande på 30 platser. Tyvärr upplevde jag att systemet med reservplatser inte funkade särskilt bra, då avhopp innan kursstart inte märks förrän den första föreläsningen, och de studenter som står i reservplats då oftast har påbörjat andra kurser istället. Detta försvårades av att studenter som inte var antagna till kursen spontant dök upp i Ladok och/eller Rapp (olika studenter i de olika systemen).

***Detta är något jag skulle önska att CSC eller KTH centralt hade bättre rutiner för.***

### Examinations- och prestationsgrad

Siffrorna är jämförbara med förra vårterminsomgången popup11. Då liksom nu var studenterna tvungna att slåss om platserna vilket antagligen ökade deras motivation (och gjorde att de som blev antagna hade mer förkunskaper).

Tyvärr är det pga problemen med Ladok och Rapp som nämns ovan inte uppenbart hur man ska räkna antal studenter. Jag har valt att räkna antal studenter som registrerat sig på kursen i Kattis, då det troligen väl överensstämmer med de som faktiskt påbörjat kursen. Detta är 29 personer. Utav dessa har en hoppat av nästan direkt. Återstående 28 studenter har hängt med genom hela kursen. Samtliga är klara med momentet LAB1 (4.5 hp), och 26 är klara med momentet ÖVN1 (4.5 hp).

Studenterna går i huvudsak tredje och fjärde året av sin utbildning. Majoriteten kommer från D-programmet men det finns även en student från SU.

### Översiktsbeskrivning av kursen

#### Kursinnehåll

Vi har följt den planering som finns på kursomgångens hemsidor. Kursen innehåller en mindre mängd teori utöver det som ingår i ADK-kursen, men förståelsen för specifika algoritmer och datastrukturer fördjupas och förmågan att använda dessa kunskaper praktiskt tränas grundligt. Tonvikten ligger på kombinatoriska problem men det förekommer även problem inom t.ex. talteori och geometri.

## Förkunskaper

Problemen kräver ofta tillämpning av algoritmer och metoder från kursen Algoritmer, datastrukturer och komplexitet (eller liknande kurs), så kursdeltagarna behöver förkunskaper motsvarande stora delar av den kursen.

## Kurslitteratur

Kurslitteraturen utgörs av ett kompendium baserat på tidigare kursomgångar. Dessutom är det lämpligt att ha en bok om algoritmer och datastrukturer; en bok som kan rekommenderas är Cormen, Leiserson, Rivest och Steins Introduction to Algorithms. Kompendiet bygger på tidigare års föreläsninganteckningar och är numera så pass komplett att det fungerar rätt bra. Det täcker dock inte in de nya ämnen som lades till i år, och dessutom finns en disjunkt uppsättning föreläsninganteckningar som i vissa fall täcker in mer än kompendiet, och i andra fall mindre.

## Undervisning

Undervisningen har dels bestått av teoriföreläsningar och dels av praktiskt orienterade övningar/labbar. Dessutom har ett antal obligatoriska problemsessioner varit en del av examinationen. De entimmesövningar som kursen innehåller har använts för att gå igenom hemtalen.

Föreläsningarna är enligt enkäten bra, men det var trots det många som slutade komma på dem. Eftersom inget tyder på att föreläsningarna stött bort studenter så beror det kanske på att en del klarar examinationen utan dem.

Mikael Goldmann har hållit tre gästföreläsningar á 90 minuter, och Lukas Polacek har hållit en gästföreläsning á 90 minuter.

Den trettonde reserv/bonus-föreläsningen ställdes in då vi redan täckt allt stoff och en e-post-förfrågan inte visade på tillräckligt starkt intresse för en bonusföreläsning.

## Examination

Kursen har två moment som vardera består av två delmoment:

### LAB1

- Problemsessioner: fyra st 4.5-timmars problemlösningsspass i labbsal
- Labbar: bygga ett litet kodbibliotek (i grupper om två)

### OVN1

- Hemuppgifter: programmeringsuppgifter som endast redovisas via Kattis (individuellt)
- Anteckningar med lösningsförslag: färdigställa anteckningar med lösningsskisser i LaTeX från en hemtalsomgång.

Problemsessionerna genomförs i grupper med två personer (i undantagsfall tre personer). Meningen är att delar av kodbiblioteket ska vara användbart på problemsessionerna.

För i stort sett varje vecka finns en uppsättning med sex hemuppgifter. Det är dock inte alls nödvändigt att lösa alla för att få godkänt på kursen. För anteckningar med lösningsförslag finns vissa krav på språk, presentation och typografi.

## Studenternas arbetsbelastning

Examinationen har varit utspridd över hela kursen. Problemsessionerna har varit vissa bestämda datum, men man har inte varit tvungen att delta vid alla tillfällen. De uppgifter som ska lösas varje vecka kan ta ganska lång tid i anspråk totalt över hela kursen om man siktar på ett bra betyg.

## Förändringar från föregående kursomgång

Trots att jag kommit in som ny kursledare är jag väl bekant med kursen sedan tidigare och har därför vågat mig på några förändringar.

1. Efter ett förslag i förra omgångens kursenkät är det nu främst studenterna som presenterar sina lösningar på hemtalsgenomgångarna (diskussion om detta nedan).
2. Syntaxanalys har tagits bort (en föreläsning), och avlusning har kortats ned till en halv föreläsning (dock, se punkt 4).
3. Datastrukturer har lagts till (en föreläsning), och vi ägnar nu en hel föreläsning åt dynamisk programmering istället för en halv.
4. Under andra halvan av kursen gjorde vi gemensam avlusning av buggig kod under ca 15 minuter varje föreläsning i syfte att träna upp just avlusningsförmågan. Mer om utfallet av detta nedan.

5. En större översyn av hemuppgifterna gjordes (vanligtvis byts ca 10-30% av uppgifterna varje omgång, nu byttes nästan alla ut).
6. Några av labbuppgifterna har bytts ut (indexmappning, miniräknare, och Catalantals-beräkning har tagits bort, och Fenwick-träd, Union-Find, och Linjära rekurrenser har lagts till)

## Enkät

En enkät har genomförts och den har besvarats av 20 personer. [Resultaten finns som bilaga.](#)

Enkäten tyder på att studenterna på det stora hela är mycket nöjda. Några svar av mer kritisk karaktär i urval med lite kommentarer:

- "Lärde mig skriva i Latex, men det är väl allt. Vad är det egentligen bra till?" (Om hemuppgiftslösningsanteckningsmomentet)
- "Det var ofta svårt att förstå när elever gick igenom sina lösningar, hade föredragit en "officiell" lösning. Jag slutade gå efter ett tag pga detta." (Om hemuppgiftsgenomgångarna)

Genomgångarna har historiskt varit ett av kursens sorgebarn, och har aldrig funkade helt bra. Årets variant, när studenterna själva presenterar lösningarna, har fått liknande omdömen (acceptabelt, men inte bra) som de lärarledda genomgångarna fått tidigare kursomgångar.

- "Resultat på labbarna snabbare, så att man vet hur man ligger till."

Detta är något jag själv kände under kursen att jag inte var nöjd med hur det funkade: både tiden för att rätta anteckningarna och tiden för att rätta labbarna var längre än jag hade önskat.

- "Ganska kul, men tyckte inte jag lärde mig särskilt mycket" (Om försöken med gemensam avlusning under andra halvan av kursen)

Kommentarerna lyfter också fram att 15 minuter inte är tillräcklig tid för att sätta sig in i koden och hitta fel, trots att det är relativt enkla program vi kollat på.

- "de fyra tillfällena när det var hemtalsdeadline torsdag lunch, labbdeadline tisdag lunch, problemsession tisdag em och sen nästa hemtalsdeadline torsdag lunch satt jag alla vakna timmar med popup." (Om arbetsbelastning)

Å ena sidan vill jag invända att det är meningen att man ska jobba med labbarna kontinuerligt under de tre veckor man har på sig att arbeta med dem, och därför inte få en så stark peak vid labbdeadline. Å andra sidan är jag fullt medveten om att världen inte riktigt fungerar på det viset.

- "Längre tid för komplettering så att man får tid att plugga till tentorna."

Sista datum för komplettering förlängdes till 20 juni från att ursprungligen ha varit 6 juni, för att inte krocka med tenta-perioderna. Jag tror att detta enkätsvar och ett liknande (som ville ha komplettering hela sommaren) inkom innan datumet förlängdes.

- "en del av föreläsningarna har gått åt till saker jag antar de flesta redan kan (saker som DP, Dijkstra m.m.)" (om ämnena som varit med i kursen)

Det fanns även några andra kommentarer som manade till ett svårare innehåll. Som kursledare är det lätt att bli entusiastmerad av sådana kommentarer och tänka "Hurra! Nästa år ska vi minsann skruva upp svårighetsgraden!", men det är ganska uppenbart att detta är fel reaktion. Som nämndes ovan var det en ovanligt stark kursomgång då det var ett tag sedan kursen gick och studenterna var tvungna att slåss om platserna (och trots det var det flera som tyckte kursen var i tyngsta laget).

Angående inkluderandet av DynProg som kommentaren nämner tycker jag det är klart motiverat: baserat på hur statistiken över lösta hemuppgifter för olika ämnen ser ut så är det många som har svårt för det. Jag vet att ADK:n nu har gjort en ganska stor omarbetning av sin DynProg-del, så det är möjligt att man kommer se en förändring av detta hos framtida studenter. Det ska bli intressant att se.

## Slutsatser om förändringar

### Anteckningar och genomgångar av hemuppgiftslösningar

I ett försök att slå två flugor i en smäll planerar jag att till nästa omgång ersätta anteckningsmomentet mot ett moment där studenterna ska presentera hemtalslösningar (detta ersätter också de befintliga hemtalsgenomgångarna). Detta kommer göras som så kallade "krysstal", där studenterna för varje hemtalsomgång får kryssa för vilka uppgifter de har förberett att presentera. Eventuellt får man dela upp studenterna i två grupper så att det finns tillräckligt många uppgifter att presentera för varje student. Min

tro/förhoppning är att kvaliteten på lösningspresentationerna kommer att öka när de blir ett obligatoriskt moment och alla deltar.

### **Gemensam avlusning**

Enkäten visar att studenterna tyckte att den gemensamma avlusningen var rolig men inte speciellt lärorik, då 15 minuter inte är tillräcklig tid för att sätta sig in i ett program och hitta dess buggar. Till nästa omgång planerar jag att stryka avlusnings-föreläsningen helt och istället sprida ut den över alla föreläsningar med ett återkommande avlusningsinslag som blir en kombination av det jag idag säger på avlusningsföreläsningen och de verkliga exempel på lösningar att avlusa som vi gjorde på den gemensamma avlusningen.

### **Labbrättnings tid**

För att underlätta att rätta labbarna snabbare ska jag till nästa omgång tänka igenom hur de ska lämnas in. I dagsläget räcker det med en Kattis-inskickning, vilket dock var otympligt för mig då jag fick ägna tid åt att lista ut vilka som hade jobbat ihop samt att Kattis inte har något bra gränssnitt för att ta fram "sista godkända inskickning innan deadline", vilket är den inskickning jag ska rätta.

### **Deadlines**

Jag ska se över interaktionen mellan hemtals-deadlines och labb-deadlines. Med nuvarande system blir det de facto så att studenterna till stor grad bara jobbar med labbarna den vecka det är deadline för labb (eftersom de alla andra veckor har hemtalsdeadlines som är mer akuta). Två möjliga lösningar är: (1) att dela upp labbarna i mindre delar och ha deadlines oftare (eller kanske snarare något bonuspoängssystem för att göra labbarna tidigare), och (2) att göra ett litet större glapp mellan hemtalsdeadlines runt varje labbdeadline (vilket i princip är att säga att vissa veckor är hemtalsveckor och andra är labbveckor -- jag är inte så förtjust i den idén och vill hellre skruva lite på examinationen på ett sådant sätt att man uppmuntras jobba med labbarna mer kontinuerligt).

### **Ämnen**

Enkätförslagen på ämnen att lägga till/ta bort är inte entydiga. Ett förslag som jag känner sympati för är att utöka och ha mer beräkningsgeometri, eftersom detta är ett ämne där studenterna ofta inte har fått lära sig mycket i tidigare kurser. Å andra sidan kan jag tycka att t.ex. Avancerade Algoritmer vore en bättre kurs att diskutera beräkningsgeometri i.

Till nästa omgång planerar jag inga ändringar från årets omgång i vilka ämnen som tas upp och i vilken omfattning utan tänker istället fokusera på de ändringar som nämns ovan.

### **Kurskompendium**

I något läge skulle jag vilja skriva ett nytt kurskompendium där fel och mindre bra passager i det existerande kompendiet fixas till, och det material som saknas (t.ex. det som lades till i kursen i år) läggs till. Detta är dock ett synnerligen tidskrävande projekt och jag vet inte när det kan tänkas bli av och när jag kan tänkas få tid för detta. Det kommer inte bli till nästa kursomgång då en stor del av min tid inför nästa läsår kommer gå åt att förbereda DD1361 som jag ska hålla till hösten.