

TDDC76 Programmering och datastrukturer, 8 hp

Kursplanering, bedömning och betygssättning

Ht 2014

Kursplanering

Timfördelning

Aktivitet	Ht1 (tim)	Ht2 (tim)	Totalt (tim)
FÖ	12	16	28
LE	8	2 + 1	10 + 1
LA	32	28	60
Summa	52	46 + 1	100 + 1

(+1 tim på LE i ht2 avser det tillfälle då utlämning av hemtentamen sker)

Schemalagd tid: 100 tim $\approx$ 7 tim/vecka

Hemarbetstid: $214 - 100 = 114$ tim $\approx$ 8 tim/vecka

Total arbetstid: 15 tim/vecka

LA i ht1 används för laborationer i C++, LA i ht2 används för projekt.

Hemarbetstiden är antalet högskolepoäng multiplicerat med 2/3 och 40 timmar, minus schemalagd tid.

Veckoplan – v.g. vänd!

Veckoplan

Laborationerna i C++ ligger i ht1, datastrukturdelen och projekt ligger i ht2, förutom framtagandet av kravspecifikation för projektet som ska göras i ht1.

Ht1		
v36	Le 1	Introduktion, C++, kompilering, enkla programexempel, begrepp
	La 1	Introduktionslaboration C++: C++-miljö på IDA, enkla program, <i>begrepp!</i>
	Fö 1	Grundläggande C++ (datatyper, deklARATIONER, uttryck, satser, funktioner, ...)
v37	Fö 2	Grundläggande C++, forts.
	La 2–3	Laboration 1.1–1.3
v38	Le 2	Uppsamling lab, genomgång laboration 1.4–5, 2, make, pekare
	La 4-5	Laboration 1.4 (klar efter La 4), 1.5 (påbörjad)
	Fö 3	Klasser
	La 6	Laboration 1.5 (klar, Laboration 1 klar)
v39	Fö 4	Klasser forts., operatoröverlagring, undantagshantering (introduktion)
	Le 3	Genomgång inför laboration 2
	La 7	Laboration 2
v 40	Lab 8-9	Laboration 2 (klar)
	Fö 5-6	Härledning/arv, polymorfi, RTTI, undantagshantering (forts.)
v 41	Le 4	Genomgång inför laboration 3
	Lab 10-12	Laboration 3
v42	La 13–16	Laboration 3 (klar)
v43-44		Omtentamensperiod vt2, tentamensperiod för ht1
Ht2 (preliminärt)		
v45	Fö 7	Objektorienterad programutveckling
	Le 5	Övning OOA/OOD, projekt
	Fö 8	Datastrukturer och algoritmer
	La 17	Projekt
v46	Fö 9–10	Datastrukturer och algoritmer
	La 18	Projekt
v47	Fö 11–13	Datastrukturer och algoritmer
	La 19	Projekt
v48	Fö 14	Datastrukturer och algoritmer
	La 20–21	Projekt (kodning)
	Le 6	Utlämnning hemtentamen, video "Sorting out Sorting"
v49-51	La 22–30	Projekt (kodning, redovisning)

Betygssättning och bedömning av laborationer, projekt och hemtentamen

Betygssättning

I kursen ingår tre examinationsmoment. **LAB2** omfattar *Laboration 1*, 2 och 3. **PRA1** är ett miniprojekt som omfattar *kravspecifikation* (KS), *analys* (OOA), *design* (OOD) och *kodning* (OOP). **UPG2** är en *hemtentamen* i datastrukturer och algoritmer som lämnas ut i fjärde veckan i ht2 och ska lämnas in måndag 11 dagar senare. Momenten betygssätts enligt följande.

LAB2	<i>Betygsätts U/3/4 (för betyg 4 krävs VG på samtliga tre delar)</i>	Deadline
	Laboration 1, G/VG	måndag 22/9, kl 13
	Laboration 2, G/VG	måndag 6/10, kl 13
	Laboration 3, G/VG	måndag 20/10, kl 13
PRA1	<i>Betygsätts i sin helhet U/G</i>	Deadline
	KS	fredag 17/10, kl 13
	OOA	tisdag 11/11, kl 13
	OOD	torsdag 20/11, kl 13
	OOP (kod, slutredovisning)	fredag 19/12
	Reflektionsdokument	mån 22/12
UPG2	<i>Betygsätts U/3/4 (motsvarar U/G/VG)</i>	måndag 8/12, kl 13

Betyg på kursen baseras på resultaten på LAB2 och UPG2. Betyg 3 på båda ger betyg 3, betyg 4 på ett av momenten ger betyg 4 och betyg 4 på båda ger betyg 5. Betyget på LAB2 och PRA1 är ett personligt betyg, vilket innebär att gruppmedlemmar kan erhålla olika betyg om det anses motiverat.

Bedömning av laborationer

Det finns en kodningsguide för C++, High Integrity C++ Coding Standard, länkad från kursens laborationssida. Denna utgör tillsammans med givna programexempel grunden för bedömning av program då det gäller kodning. Det finns även checklistor som stöd för kodning och viktiga saker att tänka på innan programkod redovisas.

Allmänna kriterier för bedömning av laborationer är **läsbarhet** – program ska enkelt kunna läsas och förstås – **korrekthet** – program ska uppfylla specifikationen – **pålitlighet** – program ska hantera felaktiga indata, felsituationer av olika slag, etc.

Läsbarhet bygger på många samverkande egenskaper, till exempel namngivning, textuell utformning, strukturering, modularisering, filuppdelning, val av språkkonstruktioner för att implementera olika slags bearbetningar, kommentering och en konsekvent och god kodningstil.

Korrekthet och **pålitlighet** kan man uppnå genom att ta fram testfall av olika slag och kontrollera att programmet producerar korrekta utdata. Testfallen ska utgöras av normala indata, felaktiga indata, och indata som utgör extremvärden i någon mening, t ex gränsvärdena för värdeintervall.

Laboration 1.1–1.3 ska lämnas in för *stilgranskning*, en gång. Laboration 1.4, 1.5, Laboration 2 och Laboration 3 ska redovisas för examination av LAB2.

För att bli *godkänd* (G) på en laboration krävs följande:

- Alla gruppmedlemmarna förutsätts till fullo medverkat i arbetet med samtliga uppgifter.
- Inga andra än gruppmedlemmarna har varit inblandade i framtagningen av koden; misstanke om annat kan bli fall för disciplinnämnden.
- Källkod ska skickas till assistenten med hjälp av skriptet ~TDDC76/send_lab.
- Deadline ska hållas, i annat fall ska assistenten kontaktas omgående.

- Muntlig redovisning är ej obligatorisk men demonstration för assistenten rekommenderas. Frågor rörande programmen ska kunna besvara av samtliga medlemmar i gruppen.
- Förutom korrekthetskravet och vad som för övrigt sägs i uppgifterna (följ specifikationerna) ska grundläggande krav på kodning uppfyllas.
- Komplettering ska lämnas in senast 3 veckor efter besked från assistenten, *utom* om man tänker satsa på VG för LAB2 då komplettering ska göras inom 8 kalenderdagar.

För att kunna erhålla betyget *väl godkänd* (VG) på en laboration krävs följande:

- Redovisning enligt ovan med samtliga VG-uppgifter lösta.
- Samtliga deadline för laborationerna ska hållas.
- Högre krav avseende kodning, korrekthet och pålitlighet än för G. *Kontrollera noga mot kodningsstandard och checklistor!*
- Det ges möjlighet att göra **en** komplettering, om man ligger *nära* VG. Sådan komplettering ska lämnas in senast 8 kalenderdagar efter meddelande från assistenten.

Alla kompletteringar bör vara inlämnade senast den dag läsperiod Vt1 startar.

Plussning, dvs höjning av betyget på LAB2 i efterhand, innebär att lösa uppgifter motsvarande det stoff som behandlas i laborationerna under datortentamensliknande form.

Bedömning av projekt

Projektet består av ett antal delmoment som successivt ska redovisas med *kravspecifikation*, *analysdokument*, *designdokument*, det framtagna *programmet* samt en *erfarenhetsrapport* (personlig eller en gruppvis sammanställning)

Dokument bedöms med hänsyn till innehåll, form och språk. De olika dokumenten ska ges en enhetlig utformning.

Program bedöms utifrån kodningsstil, och med avseende på sådant som tillkommer då man har ett större system, t ex modularisering och hur man utformat gränssnitten mellan moduler. Utfästelserna i kravspecifikationen ska uppfyllas. Användningen av C++ kommer också att ingå i bedömningen.

För att bli *godkänd* (G) på delmoment där ett dokument ska framställas krävs följande:

- Dokument ska lämnas till handledaren senast vid deadline. I annat fall ska handledaren dessförinnan kontaktas. Dokumenten är frysta i och med att handledaren mottagit dem.
- Eventuella rättelser görs i dokumentationen av nästa steg i projektet; korrigering av exempelvis analysdokumentet ska alltså göras i designdokumentet.
- Dokumenten ska uppfylla grundläggande krav på innehåll, form och språk.
- Frågor rörande dokumentens innehåll ska kunna besvaras av gruppens samtliga medlemmar.

Betyget är personligt. Varje projektmedlems insats i arbetet ska stå i proportion till gruppens storlek och projektets omfattning. Det åligger gruppen att se till att detta uppfylls, avvikelser ska rapporteras till handledaren. Dokument ska tas fram av gruppen som helhet. Kodning fördelats lämpligtvis mellan gruppens medlemmar och det ska framgå vem som skrivit vad genom kommentar i varje filhuvud.

Bedömning av hemtentamen

Samtliga uppgifter för vald betygsnivå (uppgift 1 och 2 för betyg 3, samtliga uppgifter för betyg 4) ska vara lösta, ingen uppgift eller deluppgift får hoppas över. Uppenbart oseriösa eller slarviga lösningar betraktas som olöst uppgift. Korrekt löst uppgift ger 3 poäng. Lösning med något enstaka, uppenbart slarvfel/misstag kan ge 2 poäng, annars gäller 0 eller möjligtvis 1 poäng. För godkänt (G/3) krävs 35 poäng, för väl godkänt (VG/4) krävs 51 poäng.

Det är det som behandlas på föreläsningarna och beskrivs i kursens kompendium om datastruktur och algoritmer som gäller. Eventuella alternativ och varianter som kan hittas, exempelvis på internet, är inte relevant för hemtentamen.

I fall då det skiljer mellan princip och implementering av algoritmer är det principen som gäller.