

F00 - Introduktion, datorer och lite C

Programmeringsteknik med C och Matlab, 7,5 hp

Niclas Börlin
niclas.borlin@cs.umu.se

Datavetenskap, Umeå universitet

2023-09-28 Tor

Förseningsmeddelande

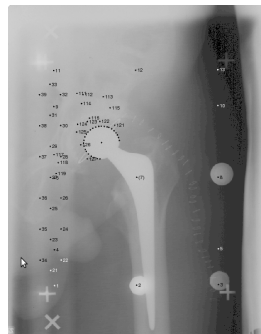
- ▶ Vid tågstationen, kan bli 5 min sen

Vad händer idag?

- ▶ Kursintroduktion
- ▶ Lite om datorer
 - ▶ Hårdvara — mjukvara
- ▶ Programvaruutveckling
 - ▶ Strukturerad problemlösning
 - ▶ Programmeringsspråket C
 - ▶ Byggstenar i C
- ▶ Efter föreläsningen
 - ▶ Datorintroduktion i labsal där ni får hjälp komma igång med verktygen ni behöver använda under kursen
 - ▶ Ni är indelade i fyra grupper C1–C4
 - ▶ Pass 1: 10–12, grupp C1–C2
 - ▶ Pass 2: 13–15, grupp C3–C4

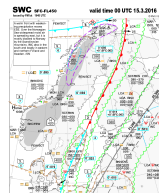
Lite om mig (1)

- ▶ Första dator
 - ▶ Commodore 64 1983 (64 kb RAM, 1MHz 6510 CPU)
 - ▶ 6510 assembler-debugger
- ▶ Doktorerade 2000
 - ▶ Mätningar i ortopediska röntgenbilder
 - ▶ Post doc 2005 Boston
- ▶ *UmRSA Digital Measure* — klinisk utvärdering av höftproteser
 - ▶ 100000+ rader Matlab
 - ▶ använts av 100+ vetenskapliga artiklar
 - ▶ 40+ universitetssjukhus, Sverige, Europa, USA, Australien, Asien



Lite om mig (2)

- ▶ Pilot, 375h
 - ▶ Privatcert 2007
 - ▶ Kommersiellt cert 2020
- ▶ Teorichef Umeå flygklubb
 - ▶ Undervisar blivande piloter i MET
 - ▶ Jobbat SMHI Arlanda
- ▶ Lärare
 - ▶ C-kurser (Datastrukturer och algoritmer snart 10 år)
 - ▶ Matlab-kurser (Icke-linjär optimering c:a 20 år)
 - ▶ Denna kurs (andra året)



Undervisningens upplägg

- ▶ **Föreläsningar** tor, fre, mån, tis 08–10, på plats UmU
- ▶ **Övningsuppgifter** publicerade på Canvas
- ▶ **Datorintroduktion** idag 10–12, 13–15
- ▶ **Handledning**
 - ▶ Normalt 10–12, 13–15 i våra lab
 - ▶ Fredag endast 13–15
 - ▶ För att alla ska få rättvis tillgång till handledning så kommer vi använda ett köverktyg som heter **tutorqueue**
- ▶ **Kodgranskning**
 - ▶ Varje torsdag 10–12 (utom idag)
 - ▶ Möjlighet diskutera **egna och andras** lösningar
 - ▶ **Få återkoppling** på egen kod
 - ▶ **Läsa andras kod** och ge återkoppling på andras kod
 - ▶ Mycket uppskattat!
- ▶ **Individuellt arbete**

Schema

V	Datum	08-10	10-12	13-15	15-17		OU1	OU2	OU3
39	23-09-28	FC	DI	DI	IA		PUB		
39	23-09-29	FC	HC	HC	IA				
40	23-10-02	FC	HC	HC	IA				
40	23-10-03	FC	HC	HC	IA				
40	23-10-04	IA	HC	HC	IA		DL0		
40	23-10-05	FC	CR	HC	IA		CR	PUB	
40	23-10-06	FC	IA	HC	IA		DL1		
41	23-10-09	FC	HC	HC	IA				
41	23-10-10	FC	HC	HC	IA		GR1		
41	23-10-11	IA	HC	HC	IA			DL0	
41	23-10-12	FC	CR	HC	IA			CR	PUB
41	23-10-13	FC	IA	HC	IA			DL1	
42	23-10-16	FC	HC	HC	IA				
42	23-10-17	FC	HC	HC	IA		DL2	GR1	
42	23-10-18	IA	HC	HC	IA				DL0
42	23-10-19	FC	CR	HC	IA				CR
42	23-10-20	IA	IA	TENTA	TENTA				
43	23-10-23	FM	FM	HM	HM				DL1
43	23-10-24	HM	HM	HM	HM		GR2	DL2	
43	23-10-25	FM	HM	HM	HM				
43	23-10-26	HM	HM	HM	HM				GR1
43	23-10-27	HM	HM	HM	HM				
44	23-10-30	TENTA	TENTA	TENTA	TENTA				
44	23-10-31	--	LP2	börjar	--				
44	23-11-01								
44	23-11-02								
44	23-11-03							GR2	DL2

FC	Föreläsning C
IA	Individuellt arbete
CR	Code review
DL1	Deadline #1
DL2	Deadline #2

DI	Datorintro C
HC	Handledning C
PUB	Lab publiceras
GR1	Rättning klar #1
GR2	Rättning klar #2

FM	Föreläsning Matlab
HM	Handledning Matlab
DL0	Inlämning för CR

Förväntade studieresultat (FSR)

- ▶ Efter avslutad kurs ska studenten...
 1. kunna tolka och beskriva **programflödet** hos program
 2. visa kännedom om grundläggande algoritmer för **sökning** och **sortering**
 3. kunna **omvandla** givna enkla **algoritmer** till C och Matlab programspråket
 4. förstå och använda sig av **variabler**, **uttryck** och **kontrollstrukturer** i ett högnivåspråk
 5. konstruera och använda **funktioner** för att skriva **strukturerade program** i C och Matlab
 6. konstruera **rekursiva funktioner** i C
 7. använda sig av grundläggande **datatyper**, **arrayer** och **strängar** samt känna till deras begränsningar
 8. visa att man förstår principerna för **felsökning** genom att kunna utföra felsökningar
 9. använda **figurer och axlar** för att presentera resultat i **Matlab/Octave**

Mål med kursen

- ▶ Personliga mål
 - ▶ Lära er **lösa problem** och skriva kod med bra **kvalité**
 - ▶ Inte bara koda *rätt*

Förändringar från ifjol

- ▶ Erfarenhet från första året
 - ▶ Ta bort material
 - ▶ Bättre överblick
- ▶ Inga bokningar i Rotundan!
- ▶ Samma labbar

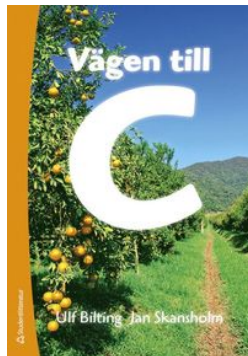
Årets kurs

- ▶ Kursens hemsida — Canvas
 - ▶ canvas.umu.se
 - ▶ Schema
 - ▶ Övningsuppgifter
 - ▶ Övrigt kursmaterial
 - ▶ Material från föreläsningar
 - ▶ Kursplan
 - ▶ Kurslitteratur mm
 - ▶ Kurssajten kommer **stängas för oregistrerade** måndag 2/10
 - ▶ Registrera er helst innan dess
 - ▶ Blir ni av med tillgången till Canvas-sidan då, registrera er **omedelbart** så återfår ni tillgången **automatiskt**
- ▶ Vi räknar med att ni kollar er epost samt besöker kurssidan **regelbundet** (dagligen)

Examination

- ▶ Moment 1
 - ▶ Digital tentamen C i skrivsal, betyg U/3/4/5
- ▶ Moment 2
 - ▶ Obligatoriska uppgifter i C, betyg U/G
- ▶ Moment 3
 - ▶ Digital tentamen i Matlab i labsal, betyg U/G
- ▶ Kursbetyg
 - ▶ Sammanfattande bedömning, betyg U/3/4/5
- ▶ Betyg
 - ▶ U — Underkänd
 - ▶ G — Godkänd
 - ▶ 3 — Godkänd
 - ▶ 4 — Icke utan beröm godkänd
 - ▶ 5 — Med beröm godkänd

- ▶ Skansholm, Bilting, *Vägen till C*, 4:e utgåvan
 - ▶ ISBN 978-91-44-07606-5, Studentlitteratur



Föreläsningar – planerat innehåll C

- ▶ Block 1:
 - ▶ F1: Kursintroduktion, datorer, program och C
 - ▶ F2: Val
 - ▶ F3: Iteration
 - ▶ F4: Repetition av föreläsning 1-3
- ▶ Block 2:
 - ▶ F5: Funktioner
 - ▶ F6: Funktioner och fält
 - ▶ F7: Enkla datatyper, pekare, fält och iteration
 - ▶ F8: Repetition av föreläsning 5-7
- ▶ Block 3:
 - ▶ F9: Datastrukturer och sökning
 - ▶ F10: Rekursion och sortering
 - ▶ F11: Pekare och strängar
 - ▶ F12: Repetition av föreläsning 9-11
- ▶ F13: Genomgång av gammal tenta
- ▶ Tenta

Obligatoriska uppgifter C

- ▶ OU 1 - FSR 3, 4 och 7 (Träna: FSR 1 och 8)
 - ▶ Inläsning från tangentbord och utskrift till skärm
 - ▶ Val
 - ▶ Repetitioner
 - ▶ Aritmetik
- ▶ OU 2 - FSR 3, 4, 5 och 7 (Träna: FSR 1 och 8)
 - ▶ Arrayer
 - ▶ Olika typer av iteration
 - ▶ Funktioner
 - ▶ Pekare
- ▶ OU 3 - FSR 3, 4, 5 och 7 (Träna: FSR 1 och 8)
 - ▶ Arrayer, iteration och funktioner
 - ▶ Egendefinierade datastrukturer

Obligatoriska uppgifter — regler

- ▶ Obligatoriska uppgifter ska lösas **enskilt**
- ▶ Obligatoriska uppgifter ska lämnas in **i tid**, även om de **inte är helt färdiga**
- ▶ Eventuella **uppskov** beslutas av kursansvarige (**restriktivt**)
- ▶ Uppgifterna bedöms enligt olika **kvalitetskriterier**, t.ex. korrekthet, tydlig kodstruktur, lättförståeliga kommentarer. . .
 - ▶ En tabell med kvalitetskriterier publiceras till **varje inlämningsuppgift**
- ▶ Får jag använda ChatGPT?
 - ▶ Ni ska **skriva** koden själva
 - ▶ Ni ska kunna **förklara** koden själva
 - ▶ Vi kommer att plocka ut $k\%$ av inlämningarna och be er **förklara och motivera** er inlämnade kod

Obligatoriska uppgifter — återkoppling kriterier

- ▶ Ni får återkoppling med **en bedömning per kriterie**
- ▶ Exempel från OU1:

Kriterium	Godkänd	Godkänd med anmärkning	Ofullständig
Testkörning	Utan fel	Mindre fel	Felaktig output Räknar fel

- ▶ Betygen på varje **kriterium** är: G, GA, O
 - ▶ Om ett kriterium får bedömningen **G** (godkänd)
 - ▶ **Klart!**
 - ▶ Om ett kriterium får bedömningen **GA** (godkänd med anmärkning)
 - ▶ Ta del av anmärkning och se till att ni **förstår** anmärkningen
 - ▶ Saker som resulterar i anmärkning på en OU kan på **senare** OU resultera i ett O
 - ▶ Om ett kriterium får bedömningen **O** (ofullständig)
 - ▶ Du har **fem arbetsdagar** på dig för att lämna in en **ny version**

Obligatoriska uppgifter — återkoppling kriterier

- ▶ Inlämningen vid DL1 kan få betygen: G, O
 - ▶ G (godkänt) om **alla** kriterier är G eller GA
 - ▶ O (ofullständig) om **något** kriterie är O
- ▶ Inlämning vid DL2 kan få betygen: G, U
 - ▶ G (godkänt) om **alla** kriterier är G eller GA
 - ▶ U (underkänt) om **något** kriterie är O
- ▶ Om en inlämning resulterar i ett **U (underkänd)**
 - ▶ Nytt försök vid **uppsamlingstillfälle** för den aktuella obligatoriska uppgiften
 - ▶ December, juni

Gruppindelning

- ▶ Under delar av kursen kommer vi att arbeta i mindre grupper
 - ▶ Datorintroduktion
 - ▶ Kodgranskningstillfällen
- ▶ Ni har delats in i grupper C1–C4 på Canvas
- ▶ Byte av grupp är ej tillåtet

Kodgranskning - C-delen

- ▶ Kodgranskningstillfällen
 - ▶ Schemalagda med handledare/lärare närvarande
 - ▶ Läsa andras kod och ge återkoppling
 - ▶ Få återkoppling och diskutera i mindre grupp
 - ▶ Fokus på kodkvalité
 - ▶ Handledarna samlar upp frågor och diskuterar i helgrupp
- ▶ Hjälpmedel
 - ▶ Bedömningskriterier – se specifikation till aktuell OU
- ▶ Tidsflöde
 - ▶ Lämna in lösning på OU senast onsdag 17.00
 - ▶ Kodgranskning torsdag
 - ▶ Eventuellt lämna in uppdaterad lösning senast fredag 17:00
 - ▶ baserat på återkoppling under kodgranskningstillfället
 - ▶ Handledarna tittar på inlämnade lösningar, gör bedömning och ger återkoppling senast tisdag 17:00

Handledningstid

- ▶ I princip **varje dag** finns handledning från våra handledare
- ▶ Handledningen kommer att vara på plats i våra **datorsalar**
- ▶ **Boka** in er för handledning via **tutorqueue** (mer om detta verktyg på datorintron)
- ▶ Antalet platser varierar — se antal lokaler i **schemat**
- ▶ Arbeta med **övningsuppgifter**
 - ▶ Innan du börjar med de obligatoriska uppgifterna
- ▶ Arbeta med de **obligatoriska inlämningsuppgifterna**
- ▶ Ta inte upp platser hela dagen om det blir platsbrist
 - ▶ Tänk på att ni är många
 - ▶ Handledarna har rätt att logga ut er om ni blockerar datorer
- ▶ Handledare kommer att ha **armband** — **gula** för C-kursen, röda för Python, bägge för Matlab

Hur man klarar kursen

1. Delta på **föreläsningar**
2. Läs i **kursboken** eller liknande
3. Gör **övningsuppgifter**
 - ▶ Gör övningsuppgifter **innan obligatorisk uppgift**
4. Gör vettigt försök till **första inlämningen**
5. Vid kodgranskningen. . .
 - ▶ Försök få så mycket information om **kvalitén på den egna koden** som möjligt
 - ▶ Försök **sätta dig in i de andras lösning** och ge dem så mycket feedback som möjligt
 - ▶ Utnyttja återkopplingen för att **förbättra** din lösning
6. Utnyttja handledarnas **kommentarer**
 - ▶ Är något oklart, **ta hjälp** av handledare eller medstudent för att förstå
7. Utnyttja **handledning** då behov finns
8. Detta är en helfartskurs! Räkna med **40 timmar** arbete per **vecka!**

Kommunikation — viktigt!

► För att få hjälp, använd i första hand:

1. Kursboken
2. Fråga på föreläsning
3. Diskussionsforum på Canvas
4. Övningsuppgifter
- 5.Handledning
6. Mail till `niclas.borlin@cs.umu.se`
7. Mail på Canvas till Niclas Börlin

Kommunikation — viktigt!

- ▶ Mail till mig kommer sällan att besvaras direkt — ni är 170 st
 - ▶ Jag svarar i stället vid nästa **föreläsning**
- ▶ Ett mail till mig ska innehålla:
 1. ert **namn**,
 2. aktuell **kurs** (5DV157 eller PCM duger bra)
 3. aktuell **läsperiod** (LP1)
 4. er **cs-användare** och
 5. ert **umu-id**!
- ▶ Saknas något av detta kanske jag raderar mailet
 - ▶ Jag kanske svarar på mailet och skriver "Id?"
- ▶ Att slå vad "Kalle Svensson" har för användarnamn är fel använd tid för mig
- ▶ Om ni skickar mail via Canvas, ange
 4. **cs-användare** och
 5. **umu-id**!