



GÖTEBORGS UNIVERSITET
IT-fakultetsnämnden

DIT151, Maskinorienterad programmering 7,5 högskolepoäng

(Machine Oriented Programming, 7.5 higher education credits)

Grundnivå/G1F

1. Fastställande

Kursplanen är fastställd 2006-11-17 av IT-universitetets fakultetsnämnd att gälla som kursplan för kurs i datalogi. Den är reviderad 2009-10-15 att gälla från höstterminen 2010.

Utbildningsområde: Naturvetenskap

Ansvarig institution: Data- och informationsteknik

2. Inplacering

Kursen ingår i Datavetenskapligt program, kandidatprogram samt som en fristående kurs vid Göteborgs universitet. Kursen är på grundnivå och har mindre än 60 hp kurs/er på grundnivå som förkunskapskrav

3. Förkunskapskrav

Kurserna DIT790 Digital- och datorteknik eller motsvarande samt DIT011 Objektorienterad programvaruutveckling eller motsvarande godkända.

4. Innehåll

Kursen innehåller moment som låter eleven konstruera större och större programblock i assembler och C. Resultatet blir ett "minirealtidssystem" för två parallella processer. Processerna styr en bormaskin och en sifferindikator för tid- och temperaturvisning.

Första delen av kursen koncentreras kring programutveckling i assemblerspråk. Därefter inhämtas grunderna för programmering i C. Slutligen kombineras programutveckling i C och Assembler.

I pedagogiskt syfte används en enkel bormaskin och traditionella in- utmatningsenheter som tangentbord och sifferindikatorer. Utrustningen används tillsammans med en mikrodator (68HC12). Allt finns tillgängligt både i form av maskinvara och som enheter i en simulator. Utrustningen används både för C- och Assemblerprogrammering.

5. Mål

De övergripande målen är att den studerande ska kunna tillämpa maskinnära programmering på inbäddade styrsystem. Målen är också att den studerande ska kunna förklara fördelar och nackdelar, möjligheter och begränsningar som karakteriserar denna klass av datorsystem.

Efter avslutad kurs förväntas studenten kunna:

a) Programutveckling i C och assemblerspråk

- Kunna utföra programmering i C och assemblerspråk samt kunna:
- Beskriva och tillämpa modularisering med hjälp av funktioner och subrutiner.
- Beskriva och tillämpa parameteröverföring till och från funktioner.
- Beskriva och tillämpa olika metoder för parameteröverföring till och från subrutiner.
- Beskriva och använda olika kontrollstrukturer.
- Beskriva och använda sammansatta datatyper (fält och poster) och enkla datatyper (naturliga tal, heltal och flyttal).

b) Programutvecklingsteknik

Att självständigt kunna:

- Konstruera, redigera och översätta (kompilera och assemblera) program
- Beskriva översättningsprocessen, dvs. assemblatorns arbetssätt, preprocessorns användning, separatkompilering och länkning.
- Testa, felsöka och rätta programkod med hjälp av avsedda verktyg.

c) Systemprogrammerarens bild av inbäddade system

Att självständigt kunna:

- Beskriva och tillämpa olika principer för överföring mellan centralenhet och kringenheter så som: ovillkorlig eller villkorlig överföring, status-test och rundfrågning.
- Konstruera program för systemstart och med stöd för avbrottshantering från olika typer av kringenheter.
- Beskriva metoder och mekanismer som är centrala i systemprogramvara så som pseudoparallell exekvering och hantering av processer.
- Beskriva och använda kretsar för tidmätning.
- Beskriva och använda kretsar för parallell respektive seriell överföring.

d) Undantagshantering i datorsystem

Att självständigt kunna:

- Beskriva och exemplifiera olika undantagstyper: interna undantag, avbrott och återstart.
- Konstruera enklare avbrottsystem med användning av digitala komponenter.
- Beskriva och tillämpa olika metoder för prioritetshantering vid multipla avbrottskällor (mjukvarubaserad och hårdvarubaserad prioritering, avbrottsmaskering, icke-maskerbara avbrott).

e) Systemets organisation

Att självständigt kunna:

- Beskriva olika typer av adressrum.
- Beskriva olika minnesorganisationer.
- Beskriva principerna för synkron, asynkron och multiplexad busskommunikation.
- Konstruera adressavkodningslogik för minneskretsar och in-/utmatningskretsar.

6. Kurslitteratur

Se separat litteraturlista.

7. Former för bedömning

Skriftlig tentamen samt godkända laborations- och programmeringsuppgifter.

Tillfälle till omtentamen ges 2 gånger per år.

Student äger rätt till byte av examinator efter att ha underkänts två gånger på samma examination, om det är praktiskt möjligt. En sådan begäran skall vara skriftlig.

I det fall en kurs har upphört eller genomgått större förändringar bör studenten garanteras tillgång till minst tre provtillfällen (inklusive ordinarie provtillfälle) under en tid av åtminstone ett år med utgångspunkt av kursens tidigare uppläggning.

8. Betyg

Betygsgraderna är Väl godkänd (VG), Godkänd (G) eller Underkänd (U). För att bli Godkänd på helkurs krävs minst 40 % poäng och för Väl Godkänd krävs minst 80% poäng av tentamens totala poängsumma. I båda fallen krävs dessutom godkända laborations- och programmeringsuppgifter.

9. Kursvärdering

Kursen utvärderas genom möten både under och efter kursen mellan lärare och studentrepresentanter. Därutöver kan en anonym enkät användas för att få skriftlig information. Resultatet av utvärderingen används för att förbättra kursen genom att visa på delar som kan läggas till, förbättras, ändras eller tas bort.

10. Övrigt

Kursen ges på svenska.