



# INSTITUTIONEN FÖR DATA- OCH INFORMATIONSTEKNIK

## DIT231 Programspråksteknik, 7,5 högskolepoäng

Programming Language Technology, 7.5 credits

*Avancerad nivå / Second Cycle*

---

### Fastställande

Kursplanen är fastställd av Institutionen för data- och informationsteknik 2012-11-12 och senast reviderad 2019-02-07. Den reviderade kursplanen gäller från och med 2019-09-02, höstterminen 2019.

*Utbildningsområde:* Naturvetenskapligt 100 %

*Ansvarig institution:* Institutionen för data- och informationsteknik

### Inplacering

Kursen ingår i Computer Science Master's Programme och ges som fristående kurs vid Göteborgs universitet.

Kursen kan ingå i följande program: 1) Datavetenskapligt program (N1COS), 2) Computer Science, Master's Programme (N2COS), 3) Software Engineering Master's Programme (N2SOM) och 4) Applied Data Science masterprogram (N2ADS)

#### *Huvudområde*

Software Engineering

Computer Science-Algorithms and Logic

#### *Fördjupning*

A1N, Avancerad nivå, har endast kurs/er på grundnivå som förkunskapskrav

A1N, Avancerad nivå, har endast kurs/er på grundnivå som förkunskapskrav

### Förkunskapskrav

Förkunskapskrav till kursen är avslutade kurser om 60 hp i ämnet Datavetenskap inklusive:

- 7,5 hp programmering (till exempel DIT143 Functional Programming, DIT953 Objektorienterad programmering och design eller motsvarande)
- 7,5 hp datastrukturer (till exempel DIT961 Datastrukturer, DIT725 Logik, algoritmer och datastrukturer eller motsvarande)

Engelska 6/Engelska B eller uppvisa motsvarande kunskapsnivå genom ett internationellt erkänt språktest till exempel TOEFL, IELTS.

### Lärandemål

Efter avslutad kurs ska studenten kunna:

#### *Kunskap och förståelse*

- förklara grundläggande implementationsproblematik för både imperativa och funktionella språk,
- förklara principerna för LL- och LR-parsning,
- förklara hur ändliga automater fungerar,

#### *Färdigheter och förmåga*

- definiera och implementera abstrakt syntax,
- definiera syntaxen för programmeringsspråk med hjälp av kontextfria grammatiker samt implementera parsrar med hjälp av standardverktyg,
- utforma regler för operativ semantik och implementera interpretatorer för både imperativa och funktionella språk,
- skriva enkla kodgeneratorer,
- tillämpa tekniken för syntaxstyrd översättning och hur man implementerar för ett valfritt programmeringsspråk,
- formulera typinferensregler och implementera typkontrollerare för funktionella och imperativa programmeringsspråk,
- utforma regler för operativ semantisk och implementera interpretatorer för funktionella och imperativa programmeringsspråk,

#### *Värderingsförmåga och förhållningssätt*

- resonera kring programspråksdesign och göra avvägningar till exempel med avseende på effektivitet och användbarhet.

### Innehåll

Syftet med kursen är att skapa förståelse för hur programmeringsspråk utformas, dokumenteras och implementeras. Kursen behandlar grundläggande tekniker och verktyg som behövs för att implementera interpretatorer (tolkar), och ger dessutom en introduktion till kompilatorer.

Studenterna kommer att lära sig programspråkens grammatik när de implementerar syntaxanalys och om typs-system vid implementering av typkontrollerare. Genom att implementera interpretatorer och kompilatorer lär sig studenten att lösa praktiska implementeringsproblem liksom teori för formell semantik.

#### *Delkurser*

1. **Tentamen** (*Written exam*), 6 hp  
Betygsskala: Väl godkänd (VG), Godkänd (G) och Underkänd (U)
2. **Laborationer** (*Laboration*), 1,5 hp  
Betygsskala: Godkänd (G) och Underkänd (U)

#### **Former för undervisning**

Undervisning ges i form av föreläsningar, övningar och laborationer med individuell handledning.

*Undervisningsspråk:* engelska

#### **Former för bedömning**

Kursen examineras genom en individuell skriftlig salstentamen i slutet av kursen och laborationer som genomförs individuellt eller i par.

Om student som underkänts två gånger på samma examinerande moment önskar byte av examinator inför nästa examinationstillfälle, ska sådan begäran inlämnas skriftligt till kursansvarig institution och bifallas om det inte finns särskilda skäl däremot (HF 6 kap § 22).

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

#### **Betyg**

På kursen ges något av betygen Väl godkänd (VG), Godkänd (G) och Underkänd (U). För att erhålla Väl godkänd (VG) för hela kursen ska studenten blivit godkänd på delmomentet laborationer och fått betyget Väl godkänd på delmomentet Skriftlig tentamen.

**Kursvärdering**

Kursen utvärderas genom möten både under och efter kursen mellan lärare och studentrepresentanter. Därutöver används en anonym enkät 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.

**Övrigt**

Kursen ersätter DIT230 Programming Languages. Kursen kan inte ingå i en examen som innehåller DIT230. Inte heller kan kursen inkluderas i en examen som baseras på en annan examen där DIT230 ingår.

Kursen kan inte ingå i en examen som innehåller DIT229. Inte heller kan kursen ingå i en examen som baseras på en annan examen där kursen DIT229 ingår.

Kursen är samläst med Chalmers.

Course literature to be announced the latest 8 weeks prior to the start of the course.