



# ID2202 Kompilatorer och exekveringsmiljöer 7,5 hp

## Compilers and Execution Environments

---

Kursplan för ID2202 gäller från och med HT10

**Betygsskala:** A, B, C, D, E, FX, F

**Utbildningsnivå:** Avancerad nivå

**Huvudområde:** -

### Lärandemål

Kursen behandlar tekniker för implementation av programmeringsspråk med hjälp av kompilatorer, både för verkliga och virtuella exekveringsmiljöer.

Det övergripande målet för kursen är att ge en förståelse för hur ett programmeringsspråk implementeras samt för de allmänna teorierna som används och hur dessa kan appliceras. Kursen kommer att behandla tekniker för att läsa, förstå, översätta, förbättra, samt exekvera program.

Efter kursen skall studenten kunna:

förklara den övergripande strukturen av en kompilator.

beskriva reguljära uttryck och ändliga tillståndsmaskiner samt använda dessa för entydig lexikal analys; kombinera och använda tekniker för att konstruera deterministiska ändliga tillståndsmaskiner från reguljära uttryck.

beskriva kontext-fria grammatiker samt använda dem för att fånga vanliga konstrukter i programmeringsspråk;

beskriva de grundläggande metoderna "top-down" samt "bottom-up" för syntaxanalys; konstruera syntaxanalyserare med hjälp av rekursiv medåkning, LL(1), LR(0), och SLR;

identifiera huruvida grammatiker kan användas med en given metod för syntaxanalys.

lista och beskriva de uppgifter som utförs under den semantiska analys-fasen.

beskriva design-principerna för intermediärrepresentationer;

förklara och använda metoder för att välja ut instruktioner;

beskriva organsieringen i ett "activation record" samt identifiera effekten av olika designval för "activation records" på programmets exekvering.

definiera aktualitet ("liveness") för variabler samt beräkna aktualitetsinformation från kontrollflödesgrafnen;

precisera principen av konservativa uppskattningar för att analysera program;

konstruera konfliktgrafer samt utföra fullständig registerallokering för program givet designen av "activation records".

namnge de viktigaste komponenterna i en exekveringsmiljö;

beskriva enkla tekniker för att hantera dynamiskt minne samt skräpsamling ("garbage collection");

identifiera den amorterade kostnaden för olika metoder av skräpsamling;

lista vanliga tekniker för virtuella exekveringsmiljöer samt förklara deras egenskaper.

beskriva vanliga tekniker för programoptimering;

identifiera loopar i program med hjälp av "dominators"; beskriva och använda tekniker för att optimera loopar och minnesåtkomster.

beskriva hur hårdvaruarkitekturers egenskaper influerar kompileringen av program;

ge exempel på viktiga egenskaper.

## Kursens huvudsakliga innehåll

- Att läsa program: lexikalanalys och syntaxanalys. Ändliga tillståndsmaskiner, reguljära uttryck, LL- och LR-parsning.
- Att förstå program: semantisk analys, typkontroll.
- Synlighetskontroll, deklarationer och uttryck.
- Att översätta program: maskiner och instruktioner.
- Intermediärkod, instruktionsval, stackar och procedur-anrop.
- Att exekvera program: virtuella exekveringsmiljöer och run-time system. Minneshantering, skräpsamling, att ladda och länka program, just-in-time kompilering. Hårdvaruarkitektursberoenden.
- Att förbättra program: optimering. Maskinoberoende optimeringar (dataflödesanalys, styrkereduktion, ...).
- Maskinnära optimeringar (registerallokering, schemaläggning, förladdning, strömförbrukning, ...).

## Undervisningsspråk

Undervisningsspråk anges i kurstillfällesinformationen i kurs- och programkatalogen.

## Behörighet

Högskolestudier motsvarande minst 120 hp inom teknik eller naturvetenskap, samt Engelska A, samt grundläggande dator teknik/arkitektur motsvarande Dator teknik grundkurs. Programmeringsvana i högnivåspråk som JAVA eller liknande. Kunskaper motsvarande kursen i Algoritmer och Datastrukturer. Kunskaper om C-programmering motsvarande de som ges i Dator teknik grundkurs.

## Litteratur

Modern Compiler Implementation in Java, Andrew W. Appel

Upplaga: Second Förlag: Cambridge University Press År: 2002

ISBN:

## Examination

- INL1 - Inlämningsuppgifter, 1,5 hp, betygsskala: P, F
- TEN1 - Tentamen, 6,0 hp, betygsskala: A, B, C, D, E, FX, F

## Krav för slutbetyg

Godkänd skriftlig tentamina (TEN1; 6hp) samt godkända inlämningsuppgifter (INL1; 1.5 hp).

Inlämningsuppgifterna betygsätts med P/F (godkänd eller underkänd). Kursen omfattar tre inlämningsuppgifter som måste lösas och lämnas in i tid (du har en veckas tid att lösa varje inlämningsuppgift). Varje inlämningsuppgift kommer att ha 6 poäng fördelat på deluppgifter. För att bli godkänd på inlämningsuppgiftsdelen av kursen måste du få minst 9 poäng totalt på de tre inlämningsuppgifterna. Om du lämnar in en uppgift i tid så kan poängen, totalt 18 poäng, även användas som bonuspoäng till tentan. Notera: bonuspoängen är endast giltiga för innevarande läsår.

Poängsumman på tentan är 200 poäng.

Betyget för kursen bestäms av det totala antalet poäng, d.v.s. summan av tentapoäng och bonuspoäng från inlämningsuppgifterna. Du behöver minst 100 poäng totalt för att bli godkänd på tentan.

Tentan betygsätts med betygen A-F. Betyget för den totala poängsumman  $n$  är som följer:

$n \geq 180$ : A

$180 > n \geq 160$ : B

$160 > n \geq 140$ : C

$140 > n \geq 120$ : D

$120 > n \geq 100$ : E

$100 > n \geq 80$ : Fx

$80 > n$  : F

I händelse av betyg Fx så är kompletterande examination möjlig inom en månad efter den ursprungliga tentan. I det fallet så kommer kursansvarige att på begäran tillhandahålla en extra inlämningsuppgift att lösas inom en veckas tid.