



# ID1004 Objektorienterad programmering 7,5 hp

Object-oriented Programming

När kurs inte längre ges har student möjlighet att examineras under ytterligare två läsår.

## Fastställande

Kursplan för ID1004 gäller från och med VT09

## Betygsskala

A, B, C, D, E, FX, F

## Utbildningsnivå

Grundnivå

## Huvudområden

Teknik

## Särskild behörighet

## Undervisningsspråk

Undervisningsspråk anges i kurstillfällesinformationen i kurs- och programkatalogen.

## Lärandemål

Du skall efter avklarad kurs kunna:

- i din datormiljö, skriva egna program, göra programmen körklara och köra dem
- ha ett angreppssätt för att attackera fel som uppstår under ovanstående process – felsökning och rättning av både syntaktiska och logiska fel
- kunna redogöra för och resonera kring den problemlösningstrategi du använt dig av eller skulle använda dig av för att lösa ett givet programmeringsproblem
- kunna söka och välja ut lämpliga programkomponenter från ett klassbibliotek
- ha grundläggande förståelse för kompilatorer och debuggers

Du visar att du uppnått dessa mål genom att muntligt redogöra för arbetsprocessen och utfallet av din körning vid arbete med och redovisning av olika programmeringsuppgifter.

- egenhändigt angripa och lösa en given programmeringsuppgift
- muntligt redogöra för och tillämpa en problemlösningstrategi och motivera de egna valen
- använda dig av relevanta programstrukturer (metoder, klasser etc.) för att underlätta lösandet av programmeringsuppgifter
- förvissa dig om att lösningen uppfyller givna krav (testning)
- kunna använda de grundläggande funktionerna i en debugger
- kunna använda dig av existerande programkomponenter
- kunna resonera kring ett programs kvalitativa aspekter (till exempel med avseende på tidskomplexitet, modularisering, med mera)

Du visar att du nått dessa mål genom att lösa, skriftligt dokumentera och muntligt redovisa inlämningsuppgifterna.

- redogöra för grundläggande datalogiska begrepp och relationer mellan dessa begrepp (selektion, iteration, datatyper, variabler, etc.)
- redogöra för och resonera kring grundläggande objektorienteringsbegrepp (objekt, klass, inkapsling etc.)
- utifrån en beskrivning av ett mindre problem kunna skriva en algoritm som löser problemet
- illustrera och förklara en algoritm i en algoritmnotation (till exempel pseudokod)
- kunna resonera kring ett programs kvalitativa aspekter (till exempel med avseende på tidskomplexitet, modularisering, med mera)

Du visar att du uppnått dessa mål på tentamen.

## Kursinnehåll

## Kurslitteratur

Preliminärt:

John Lewis & William Loftus: Java Software Solutions (Edition/upplaga: 6), Addison Wesley, 2009

Ett kompendium med föreläsningmaterial, inlämningsuppgifter, etc finns på kårbokhandeln från och med ungefär en vecka före kursstart.

En kortversion av densamma, som innehåller allt man absolut behöver veta, kommer att mailas ut till alla registrerade studenter ungefär en vecka före kursstart. Därefter hittar man detta dokument i kursens Moodle-modul.

# Examination

- INL1 - Inlämningsuppgifter, 4,5 hp, betygsskala: P, F
- TEN1 - Tentamen, 3,0 hp, betygsskala: A, B, C, D, E, FX, F

Examinator beslutar, baserat på rekommendation från KTH:s handläggare av stöd till studenter med funktionsnedsättning, om eventuell anpassad examination för studenter med dokumenterad, varaktig funktionsnedsättning.

Examinator får medge annan examinationsform vid omexamination av enstaka studenter.

## Inlämningsuppgifter

Skriftlig eller muntlig tenta

Betyget bestäms av tentan, med lite hjälp av inlämningsuppgifterna. Tentan är organisatoriskt uppdelad i tre delar. En del examinerar faktakunskaper och examineras kontinuerligt under kursens gång. Denna del är i sin tur indelad i mindre avsnitt som behandlar var sitt ämne. Vissa av dessa avsnitt är obligatoriska för att bli godkänd, andra krävs bara för högre betyg.

Den andra delen av tentan examinerar förståelsen och att man kan tillämpa teorin i praktisk programmering. Denna del görs i tentaform, men med tillgång till datorer. Denna del är obligatorisk för ett godkänt betyg.

Den sista delen av tentan består av en traditionell salstenta. Denna del är bara nödvändig för högre betyg.

För samtliga godkända betyg måste kraven för eventuella lägre godkända betyg också vara uppfyllda.

A. Algoritmen är logiskt felfri och den objektorienterade modellen modellerar det givna problemet på ett i princip felfritt sätt.

B. Grunden för både algoritm och modell är korrekt, men det får finnas enstaka mindre brister i utförandet, till exempel något missat specialfall.

C. Man klarar av att för ett givet problem utforma och implementera en algoritm som löser en väsentlig del av problemet, att specialfall inte behandlas kan accepteras. Modellen hänger logiskt samman.

D. Godkänd på samtliga avsnitt på faktadelen av tentan. Arv, inkapsling och överskuggning används, på ett för problemet, korrekt sätt i modellen.

E. Godkänt på samtliga obligatoriska avsnitt av faktadelen. Man klarar av att omvandla en given algoritm till (pseudo-) kod. Man klarar av att konstruera en objektorienterad modell över åtminstone de viktigaste dataklasserna för ett givet problem och kan visa hur dessa skulle kunna användas i ett program. Modellen måste vara korrekt i förhållande till problemet, men behöver inte vara heltäckande.

Fx. Godkänt på samtliga obligatoriska avsnitt av faktadelen. Man klarar av att omvandla en given algoritm till (pseudo-) kod. Man klarar av att konstruera en objektorienterad modell över åtminstone de viktigaste dataklasserna för ett givet problem.

## Övriga krav för slutbetyg

Godkänt på tentamen samt inlämningsuppgifter.

## Etiskt förhållningssätt

- Vid grupparbete har alla i gruppen ansvar för gruppens arbete.

- Vid examination ska varje student ärligt redovisa hjälp som erhållits och källor som använts.
- Vid muntlig examination ska varje student kunna redogöra för hela uppgiften och hela lösningen.