



# ID1218 Tillämpad programmering 7,5 hp

Applied Programming

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

## Fastställande

Kursplan för ID1218 gäller från och med HT08

## Betygsskala

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

## Utbildningsnivå

Grundnivå

## Huvudområden

Teknik

## Särskild behörighet

**Grundläggande behörighet samt Matematik D, Fysik B och Kemi A**

## Undervisningsspråk

Undervisningsspråk anges i kurstillfällesinformationen i kurs- och programkatalogen.

# Lärandemål

Kursens allmänna mål är att förmedla den nödvändiga kunskapen inom programmerings-modeller, -koncept, -språk, och -miljöer som behövs i modern mjukvarukonstruktion. Några få exempel på applikationsområden är att skapa webb-baserade, inbäddade, eller distribuerade system.

Efter fullgjord kurs skall teknologen:

- känna till grundläggande egenskaper och centrala koncept inom vanligt förekommande programmeringsmodeller (objekt-orienterad, imperativ, funktionell, och parallell programmering) samt känna till vanliga programmeringsspråk som implementerar dessa modeller.
- analysera programmeringsuppgifter och identifiera aspekter som gör dem lämpade för de ovan nämnda programmeringsmodellerna.
- kunna tillämpa objekt-orienterade koncept från Java i C++: klasser, arv, objekt-baserad polymorfism, samt undantag.
- förstå och kunna tillämpa koncept från C++ som implementerar programmeringsmodeller för objekt-orienterad och imperativ programmering: parameter-överföring, pekare och referenser, explicit minneshantering, överlagring, parametrisk polymorfism via mallar (templates).
- kunna konfigurera, kompilera, samt använda sig av mjukvarubibliotek skrivna i C++.
- kunna designa, strukturera, debugga, testa, samt profilera mellanstora program skrivna i C++.
- kunna tillämpa grundläggande funktionella programmeringstekniker i Erlang: rekursion, "pattern matching", listhantering.
- förstå och kunna identifiera "safety" och "liveness" egenskaper hos parallella program.
- kunna tillämpa "mutual exclusion", "message ordering", och "concurrency adaptor" som vanligt förekommande mönster för synkronisering i parallella program.
- kunna strukturera parallella applikationer i form av kommunicerande och samarbetande parallella komponenter.

# Kursinnehåll

- Programmeringsmodeller och -språk: imperativ programmering, objekt-orienterad programmering, funktionell programmering, parallell programmering. Introduktion till centrala egenskaper, fördelar, samt typiska applikationsområden.

- Programmering i C++: i jämförelse till Java. Centrala skillnader. Relation till C. Grunder: datatyper, kontrollstrukturer, objekt, klasser, arv, objekt-baserad polymorfism. Avancerat: parameter-överföring, referenser, pekare och explicit minneshantering, parametrisk polymorfism via mallar (templates).

- Parallellprogrammering: paradigm (delade datastrukturer kontra meddelandesändning), tekniker, och koncept ("safety", "liveness", "concurrency control", tjänster).

- Funktionell- och parallell-programmering i Erlang: rekursion, listhantering, "pattern matching", meddelandesänding, "safety", ("mutual exclusion"), "liveness" ("concurrency adaptor", "message ordering").

- Pragmatiska aspekter: utvecklings-verktyg och -miljöer, att testa, debugga, och profilera.

## Kurslitteratur

Mark Allen Weiss, C++ for Java Programmers, Prentice Hall, 2004.

Concurrent Programming in Erlang, Prentice Hall, 2001 (elektronisk version).

## Examination

- LAB1 - Laborationer, 3,0 hp, betygsskala: P, F
- TEN1 - Tentamen, 4,5 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.

## Övriga krav för slutbetyg

Godkänd tentamen (TEN1; 4.5 hp) samt godkända laborationsuppgifter (LAB1; 3 hp).

## 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.