



Objektorienterad mjukvaruutveckling

Object-oriented Software Development

15 högskolepoäng

15 credits

Ladokkod: 21001B

Version: 2.0

Fastställd av: Utskottet för utbildningar inom bibliotek, information och IT 2016-02-02

Gäller från: HT 2016

Nivå: Grundnivå

Huvudområde (successiv fördjupning): Informatik (G1F)

Utbildningsområde: Naturvetenskap

Ämnesgrupp: Informatik/Data- och systemvetenskap

Förkunskapskrav: Avklarad kurs Data- och Programstrukturer 7,5 högskolepoäng eller motsvarande

Betygsskala: Underkänd, Godkänd eller Väl godkänd

Innehåll

Kursen har två spår som löper delvis parallellt. Det inledande spåret fokuserar på objektorienterad programmering i Java. Detta spår inleds med en ingående genomgång av grunderna för objektorienterad programmering till exempel klasser, inkapsling, arv samt polymorfism. Därefter behandlas mer språkspecifika moment såsom undantagshantering, händelsehantering, layoutmanagers, trådning, nätverkskommunikation och databaskommunikation. I undervisningen relateras designmönster som ingår i GRASP och GoF till programmeringen.

Det andra spåret handlar om objektorienterad mjukvaruutveckling. Detta spår behandlar grunderna inom objektorienterade utvecklingsmetoder och -filosofier, objektorienterad analys och design samt testning. Undervisningen om objektorienterad analys och design utgår ifrån Unified Process (UP). Inom ramen för det andra spåret genomförs ett projekt som inbegriper analys, design, konstruktion och testning av en lösning för ett komplexare problem.

Mål

Efter avklarad kurs ska studenten kunna, med avseende på,

Kunskap och förståelse

- 1.1 ingående redogöra för de viktigaste principerna för objektorienterad programmering,
- 1.2 ingående redogöra för de viktigaste principerna inom objektorienterad analys och design,
- 1.3 redogöra för grundläggande principer inom testning,
- 1.4 ingående redogöra för utvecklingsmetoder, -modeller och -filosofier inom objektorientering,
- 1.5 redogöra för utvecklingsmetoder, -modeller och -filosofier inom mjukvaruutveckling,
- 1.6 ingående redogöra för nyttan och innebörden av designmönster,
- 1.7 redogöra för och beskriva designmönster som ingår i GRASP och GoF,

Färdighet och förmåga

- 2.1 använda de vanligaste konstruktionerna i Java och dess standardbibliotek,
- 2.2 använda konstruktionerna för undantagshantering i Java och dess standardbibliotek,
- 2.3 använda konstruktionerna för händelsehantering i Java och dess standardbibliotek,
- 2.4 använda konstruktionerna för gränssnittskonstruktion i Java och dess standardbibliotek,
- 2.5 använda konstruktionerna för trådning i Java och dess standardbibliotek,
- 2.6 använda konstruktionerna för databashantering i Java och dess standardbibliotek,
- 2.7 använda konstruktionerna för nätverkskommunikation i Java och dess standardbibliotek,
- 2.8 utföra grundläggande modellering som en del av den objektorienterade analysen och designen,
- 2.9 genomföra en objektorienterad analys och design av komplexare problem,
- 2.10 använda testning vid genomförande av objektorienterad utveckling,

- 2.11 tillämpa och implementera designmönster i ett objektorienterat programspråk,
- 2.12 konstruera större objektorienterade datorprogram enligt goda objektorienterade principer
- 2.13 muntligt och skriftligt redogöra för olika utvecklingsmetoder, -modeller och -filosofier samt

Värderingsförmåga och förhållningssätt

- 3.1 välja ett lämpligt designmönster för en specifik situation.

Undervisningsformer

Undervisningen bedrivs på svenska i form av föreläsningar, laborationer, projekt, skriftliga inlämningsuppgifter, handledning samt seminarier.

Examinationsformer

Kursen examineras genom följande examinationsmoment:

Teoretisk tentamen:

Lärandemål: 1.1-1.7, 3.1

Högskolepoäng: 2.0

Betygskala: UVG

Praktisk tentamen:

Lärandemål: 2.1-2.8

Högskolepoäng: 3.0

Betygskala: UVG

Projekt: Större utvecklingsarbete i grupp

Lärandemål: 2.1-2.12, 3.1

Högskolepoäng: 5

Betygskala: UVG

Laboration 1: Programmeringsuppgift i grupp

Lärandemål: 2.1-2.2, 2.8

Högskolepoäng: 1.5

Betygskala: UG

Laboration 2: Programmeringsuppgift i grupp

Lärandemål: 2.1-2.5, 2.8, 2.11

Högskolepoäng: 1.5

Betygskala: UG

Inlämningsuppgift: Individuell inlämningsuppgift

Lärandemål: 1.4-1.5, 2.13

Högskolepoäng: 2

Betygskala: UG

Kursen bedöms med betygen U/G/VG. Projektet och tentamen poängsätts och för att erhålla betyget väl godkänt på hel kurs krävs att summan av projektets och tentamens poäng överstiger 75% av maxpoängen.

Då kursplanen ändras kommer student som önskar slutföra rester från ett kurstillfälle att examineras utifrån kursens nya innehåll och upplägg. Då kursen har upphört kan student som önskar slutföra rester följa hela eller delar av annan likvärdig kurs.

Studentens rättigheter och skyldigheter vid examination är enligt riktlinjer och regelverk vid Högskolan i Borås.

Kurslitteratur och övriga läromedel

Kurslitteraturen är på engelska.

Liang, Yiang D. (Senaste upplagan, 2013 eller senare). Introduction to Java programming: comprehensive version. Harlow: Pearson Education.

Larman, Craig. (Senaste upplagan, 2005 eller senare). Applying UML and patterns: an introduction to object-oriented analysis and design and iterative development. Upper Saddle River, N.J.: Prentice Hall PTR.

Studentinflytande och utvärdering

Kursen utvärderas i enlighet med gällande riktlinjer för kursvärderingar vid Högskolan i Borås, där studenternas synpunkter ska inhämtas. Kursutvärderingsrapporten publiceras och återkopplas till deltagande och blivande studenter i enlighet med ovan nämnda riktlinjer, och ligger till grund för framtida utveckling av kurser och utbildningsprogram. Kursansvarig lärare ansvarar för att utvärdering enligt ovan genomförs.

Övrigt

Kursen ges inom Systemarkitekturutbildningen.