

Kursplan för

Multicoreprogrammering Multicore Programming

**EDAN26, 7,5 högskolepoäng, A
(Avancerad nivå)**

Gäller för: Läsåret 2022/23

Fakultet: Lunds tekniska högskola

Beslutad av: Programledning C/D

Beslutsdatum: 2022-04-07

Allmänna uppgifter

Valfri för: C5-pv, D5-is, D5-pv, E4-pv, F4, F4-pv, Pi4-pv

Undervisningsspråk: Kursen ges på svenska

Syfte

Kursens syfte är att studenterna ska lära sig att implementera en algoritm så effektivt som möjligt på en given multicoredator, med fokus på multiprocessorer med delat minne. Ett annat syfte är att få insikt om fördelar och nackdelar med olika språk för multicore programmering, såsom Java, Scala, C samt OpenMP. Studenterna ska även få förståelse för låsningsfria algoritmer och datastrukturer.

Mål

Kunskap och förståelse

För godkänd kurs skall studenten

- förstå hur cacheminnen fungerar på multicores
- förstå vikten av att minska antal cachemissar
- förstå vikten av load-balancing mellan olika processorer
- förstå hur man kan dela upp ett sekventiellt C program med hänsyn tagen till ovanstående
- förstå fördelar och nackdelar med olika multicorearkitekturer dels med avseende på (1) prestanda och dels på (2) hur komplicerad programmeringen blir
- förstå fördelar och nackdelar med olika programmeringsspråk

- förstå när låsfria algoritmer är lämpliga

Färdighet och förmåga

För godkänd kurs skall studenten

- kunna använda Pthreads för att skapa trådar och synkronisera dessa på en multiprocessor
- kunna använda OpenMP för att parallellisera beräkningsintensiva sekventiella C program
- kunna förbättra prestandan på ett givet sekventiellt program genom att parallellisera detta på ett sätt som maximerar prestandan.
- kunna välja programspråk för ett givet problem

Värderingsförmåga och förhållningssätt

För godkänd kurs skall studenten

- kunna hitta en fördelaktig version av en algoritm med hänsyn taget till tillämpningens krav och arkitekturen hos den aktuella multiprocessorn som ska användas
- kunna välja rätt multicorearkitektur för en given algoritm.
- kunna välja rätt programspråk för en given algoritm.

Kursinnehåll

Multiprocessorer, klassificering av cachemissar, parallellisering av ett sekventiellt program genom decomposition, assignment, orchestration, och mapping, owner-computes regeln, memory consistency models, sequential consistency, weak ordering, release consistency, Pthreads, OpenMP, aktuell forskning inom multicores.

Kursens examination

Betygsskala: TH - (U,3,4,5) - (Underkänd, Tre, Fyra, Fem)

Prestationsbedömning: Muntlig tentamen. För godkänt betyg krävs godkänd tentamen, godkända datorlaborationer samt godkänd inlämningsuppgift (projekt). Projekt utgörs av ett program som ska skrivas för en multiprocessor med fyra processorer. Slutbetyg i kursen grundar sig på resultatet av den muntliga tentamen.

Om så krävs för att en student med varaktig funktionsnedsättning ska ges ett likvärdigt examinationsalternativ jämfört med en student utan funktionsnedsättning, så kan examinator efter samråd med universitetets avdelning för pedagogiskt stöd fatta beslut om alternativ examinationsform för berörd student.

Delmoment

Kod: 0122. **Benämning:** Obligatoriska moment.

Antal högskolepoäng: 3,5. **Betygsskala:** UG. **Prestationsbedömning:** Godkända obligatoriska moment. **Delmomentet omfattar:** Laborationer och ett projekt.

Kod: 0222. **Benämning:** Tentamen.

Antal högskolepoäng: 4. **Betygsskala:** TH. **Prestationsbedömning:** Godkänd tentamen. **Delmomentet omfattar:** Muntlig tentamen.

Antagningsuppgifter

Förkunskapskrav:

- EDAA01 Programmeringsteknik - fördjupningskurs eller EDAA30 Programmering i Java - fortsättningskurs

Begränsat antal platser: Nej

Kursen överlappar följande kurser: EDA116, EDAN25

Kurslitteratur

- Jonas Skeppstedt and Christian Söderberg: Writing Efficient C Code: A Thorough Introduction, 3rd edition. Skeppberg, 2019, ISBN: 9781723831157.

Kontaktinfo och övrigt

Kursansvarig: Jonas Skeppstedt, jonas.skeppstedt@cs.lth.se

Hemsida: <http://cs.lth.se>