



KURSPLAN

Multiprocessorprogrammering Multiprocessor Programming 6 högskolepoäng (6 credits)

Kurskod: DV2606

Huvudområde: Datavetenskap

Utbildningsområde: Teknik

Utbildningsnivå: Avancerad nivå

Fördjupning: AIN - Avancerad nivå, har endast kurs/er på grundnivå som förkunskapskrav

Undervisningsspråk: Engelska

Gäller från: 2021-08-30

Fastställd: 2021-03-01

1. Beslut

Denna kurs är inrättad av dekan 2021-02-03. Kursplanen är fastställd av prefekten vid institutionen för datavetenskap 2021-03-01 och gäller från 2021-08-30.

2. Förkunskapskrav

För tillträde till kursen krävs 60 hp avklarade i huvudområdet datavetenskap eller programvaruteknik inklusive avklarade kurser i programmering 12 hp, algoritmer och datastrukturer 6 hp, operativsystem samt datorteknik 6 hp.

3. Syfte och innehåll

3.1 Syfte

Parallellism har under många år varit ett sätt att uppnå hög prestanda i datorsystem. Datorsystem består idag av parallellism på flera olika nivåer: regelbundna arrayer (matriser) av SIMD processorkärnor som i GPU:er, ett mindre antal generella processorkärnor som i multicore-processorer, eller sammankopplade noder som i ett distribuerat system. För att kunna utnyttja hårdvaran till fullo, behöver man skriva effektiva parallella program. Mao, parallella datorsystem och parallell programmering är fundamentala grundstenar för dagens och framtidens datorer.

Denna kurs syftar till att studenterna ska lära sig att programmera datorsystem som innehåller många processorer eller processorkärnor så att man uppnår hög prestanda. Vi kommer att gå igenom programmering för delat minne, både MIMD och SIMD, samt programmering av distribuerade system. Fokus i kursen kommer att ligga på programmering av multiprocessorer med delat minne.

3.2 Innehåll

Kursen omfattar följande moment:

- Introduktion till multiprocessorer och parallell programmering.
- Konstruktionsprinciper för parallella datorarkitekturer och datorsystem.
- Programmeringsmodeller och konstruktionsprinciper för parallella program för att uppnå hög prestanda, t ex uppdelning av arbetet för att passa underliggande arkitekturer, kommunikation och synkronisering.
- Bibliotek för parallell programmering, såsom t ex pthreads och CUDA.
- Exempelalgoritmer för ett antal intressanta applikationsområden.
- Praktisk övning i utveckling av parallella program.

4. Lärandemål

Följande lärandemål examineras i kursen:

4.1 Kunskap och förståelse

Efter genomförd kurs ska studenten kunna:

- utförligt förklara uppbyggnad och funktion hos olika typer av multiprocessorer och parallella datorsystem
- utförligt förklara olika typer av programmeringsmodeller för multiprocessorer och parallella datorsystem
- utförligt förklara olika principer för att designa parallella program för att få hög prestanda samt den underliggande arkitekturs inverkan på prestandan

4.2 Färdighet och förmåga

Efter genomförd kurs ska studenten kunna:

- tillämpa olika principer och tekniker för att utveckla och implementera parallella program
- analysera dataflödet i problem för att kunna partitionera det i parallella delar
- utföra enklare prestandamätningar och prestandaanalys av parallella program
- utföra lämpliga optimeringar i parallella program, t ex genom att minska kommunikations- och synkroniseringsbehov, i syfte att uppnå hög prestanda

4.3 Värderingsförmåga och förhållningssätt

Efter genomförd kurs ska studenten kunna:

- skriftligt och muntligt redogöra för och motivera sina lösningar
- självständigt och kritiskt utvärdera och analysera sina lösningar

5. Läraktiviteter

Undervisningen är en kombination av föreläsningar och laborationer. Föreläsningarna presenterar och förklarar de teoretiska grunderna. Studenten förväntas dessutom att självständigt tillgodogöra sig teoretiska kunskaper med hjälp av självstudier av relevant litteratur. Laborationerna fokuserar på övningar som utvecklar studentens kunskap inom området.

De obligatoriska laborationsuppgifterna genomförs enskilt eller i grupp inom givna tidsramar. Om laborationer kan genomföras i grupp anges det vid kursstart och i kurs-PM. Laborationsuppgifterna skall genomföras huvudsakligen i programmeringsspråken C / C++. Studenten förväntas ha tillräcklig förståelse för språken för att självständigt klara av att omvandla teori till praktik för lösa laborationsuppgifterna.

6. Bedömning och examination

Examinationsmoment för kursen

Kod	Benämning	Omfattning	Betyg
2020	Laboration 1	1,5 hp	GU
2030	Laboration 2	1,5 hp	GU
2110	Salstentamen[1]	3 hp	AF

[1] Bestämmer kursens slutbetyg vilket utfärdas först när samtliga moment godkännts.

Kursen bedöms med betygen A Utmärkt, B Mycket bra, C Bra, D Tillfredsställande, E Tillräckligt, FX Underkänd, något mer arbete krävs, F Underkänd.

I kursstillfallets information inför kursstart framgår i vilka examinationsmoment som kursens lärandemål examineras samt gällande bedömningsgrunder.

Examinator kan, efter samråd med högskolans FUNKA-samordnare, fatta beslut om anpassad examinationsform för att en student med varaktig funktionsvariation ska ges en likvärdig examination jämfört med en student utan funktionsvariation.

7. Kursvärdering

Kursvärdering ska göras i enlighet med BTH:s beslut om frågeställning i kursvärderingar och beslut om process för hantering och uppföljning av kursvärderingar.

8. Begränsningar i examen

Kursen kan ingå i examen men inte tillsammans med annan kurs vars innehåll, helt eller delvis, överensstämmer med innehållet i denna kurs.

9. Kurslitteratur och övriga lärresurser

Huvudlitteratur

David B. Kirk, Wen-mei W. Hwu, Programming Massively Parallel Processors: A Hands-on Approach, Third Edition, 2016 ISBN-9780128119860 (paper back) ISBN-9780128119877 (ebook).

T. Rauber and G. Rünger, Parallel Programming for Multicore and Cluster Systems, 2nd ed., Springer, 2013, ISBN 978-3-642-37800-3.

Samt material från institutionen.

Referenslitteratur

Titel: Wolfgang Engel, GPU PRO 4: Advanced Rendering Techniques, ISBN-10: 1466567430

ISBN-13: 978-1466567436.

Jason Zink, Matt Pettineo, Jack Hoxley, Practical Rendering and Computation with Direct3D 11, ISBN-10: 1568817207, ISBN-13: 978-1568817200.

A. Grama, A. Gupta, G. Karypis, and V. Kumar, Introduction to Parallel Computing, 2nd edition, Addison-Wesley, 2003, ISBN 0201648652.

J.L. Hennessy and D.A. Patterson, Computer Architecture – A Quantitative Approach, 4th edition, Morgan

Kaufmann Publishers, 2007, ISBN 0-12-370-490-1.

Maurice Herlihy, Nir Shavit, Victor Luchangco, and Michael Spear, The Art of Multiprocessor Programming, 2nd edition, Morgan Kaufmann Publishers, 2020, Paperback ISBN: 9780124159501, eBook ISBN: 9780123914064.