Graduate Co-op Students Information Manual

Department of Computer Science

Faculty of Science

University of Regina

2014

Table of Contents

1. Department Description

The Department of Computer Science offers programs of study involving interdepartmental, multi-institutional and inter-institutional collaboration that has attracted faculty members and graduate students from all over the world. Students may pursue full-time or part-time graduate study leading towards the MSc and PhD degrees.

The MSc and PhD degrees in Computer Science focus on four main areas of research: artificial intelligence; databases and information retrieval; graphics, image and audio processing; multimedia, and software engineering. Specifically, active research topics conducted by faculty members include, but are not limited to:

- Data mining, knowledge discovery and machine learning, Bayesian networks, rough set theory, uncertainty management, quantitative and temporal reasoning
- Graphical modeling and rendering algorithms, animation, image and signal processing, facial recognition, computational music and audio, information visualization, human computer interaction
- Information retrieval, cognitive informatics, Web intelligence and service, electronic commerce, database theory, and information theory and its applications in communications
- Language-based software security, data security, agent-oriented software engineering, software reuse, formal methods, and distance education
- Algorithm design and analysis, theory of computing, computational geometry, graph theory

The Department of Computer Science maintains several research laboratories: Animation Software Design, Artificial Intelligence, Graphics, Intelligent Database System, Interactive Media, Computational Discovery, Multimedia Gaming, Open Systems, Rough Computing, Rough Music and Audio Digital Interactive (aRMADILo), Saskatchewan Research Network Digital Media, Software Engineering, and Web Intelligence. Both the TR and New Media Studio laboratories result from collaborative research with various partners from industry, university, and government.

For detailed information about the research interests of faculty members and ongoing research of graduate students, please visit the Department's website at http://www.cs.uregina.ca/.

2. Program Requirements and Procedures

Co-op option

The Master's co-op option requires students to pursue research areas supported by the Department of Computer Science. A fully qualified student may complete a Master's co-op option by undertaking 21 credits of coursework; 12 credits of professionally oriented work terms (CS601 and CS602); and 3 credits of co-op education placement project report, presentation and defense (CS600).

MSc - Co-op (36 credits)

CS 8xx	3 cr hrs
CS 8xx	3 cr hrs
CS 8xx	3 cr hrs
CS 8xx	3 cr hrs
CS*	3 cr hrs
CS/non-CS*	3 cr hrs
CS/non-CS*	3 cr hrs
CS 601	6 cr hrs
CS 602	6 cr hrs
CS 600	3 cr hrs
TOTAL	**36 cr hrs**

* one of these may be a 400-level class (others are 800-level)

NB:

- At least ONE CS800-level course must be completed at the University of Regina before a student starts to apply for the first work term.
- At least THREE CS 800-level courses must be completed at the University of Regina as part of the course requirements, although we allow students to take courses outside CS, and a 400-level course, and transfer credits in from elsewhere.

2.1 Pre-Approved Non-Computer Science Courses

The following courses have been pre-approved and will satisfy the non-CS course requirement in all CS graduate programs listed above. **Please note** that this is not meant to be an exhaustive list of the non-CS courses that may be taken. Its only purpose is to itemize those courses that have already been examined and approved. Students are encouraged to consider courses not on the list that are relevant to their programs, while being reminded that all non-CS courses not on the list must be approved. Please see the relative programs areas on the FGSR website for course descriptions.

Courses in Electronic Systems Engineering:
ENEL 489, 492, 495, 811, 812, 813, 850, 857

Courses in Software Systems Engineering:
ENSE 483, 882

Courses in Mathematics and Statistics:
MATH 809, 827, 869, STAT 852, 871

Courses in Business Administration:
GBUS 866

2.2 Course Descriptions

CS 600 Graduate Co-op Report (3)
This is a 3 credit hours course. The student writes a formal technical work term report and makes a formal presentation of the report.

CS 601 Graduate Co-op Work Term (6)
First work placement.

CS 602 Graduate Co-op Work Term (6)
Second work placement.

CS 802 Analysis and Design of Parallel Algorithms (3)
Theoretical and practical aspects of parallel algorithms; functional descriptions of various parallel models of computations; interconnection networks for multi-computers. Prior to registering in this course, students should have a background in parallel computing comparable to the senior undergraduate level.

CS 805 Computer Graphics (3)
Geometric and other advanced modeling techniques; image rendering and synthesis techniques; interactive graphics; issues in computer animation. Prior to registering in this course, students should have a background in computer graphics comparable to the senior undergraduate level.

CS 807 Interactive Hardware and Embedded Computing (3)
Hardware design for physical and pervasive computing systems. Wired and wireless communication protocols; sensors and actuators; resource constraints; location- and context - awareness. Applications include wearable computing, wireless sensor networks, robotics and automation, internet of things. Embedded hardware platforms such as ARM (raspberry Pi) and AVR (Arduino).

CS 808 Advanced Animation Software Design (3)
Principles of animation. Current research areas in animation software design. Features and architecture of animation software. Timelines, motion pathways, parametric key framing kinematics, gaseous phenomena, and facial animation.

CS 809 Interactive Entertainment Software (3)
This course surveys current research on the design and implementation of interactive entertainment software, including computer games. Topics include: interactivity, principles of interactive entertainment, hardware platforms, current software development tools and languages, game loop, design of virtual worlds and virtual characters, real-time requirements, incorporating multimedia resources, aesthetics.

CS 811 Theory of Computing (3)
Study of fundamental concepts of computer science from the theoretical point of view; basic concepts of computational complexity theory, algorithm analysis and their relation to the set of problems which can be programmed; "good" algorithm design. Prior to registering in this course, students should have a background in introductory compiler design, or algorithm analysis comparable to the senior undergraduate level.

CS 815 Computer Vision (3)
Sensing techniques; sensing data pre-processing; higher level scene descriptions; model-based recognition; motion analysis. Prior to registering in this course, students should have a background in image processing comparable to the senior undergraduate level.

CS 820 Artificial Intelligence (3)
Logics; natural language processing; knowledge representation; uncertainty reasoning; machine learning; expert systems; neural networks. Prior to registering in this course, students should have a background in artificial intelligence comparable to the senior undergraduate level.

CS 824 Information Retrieval (3)
Content analysis; types of storage and retrieval systems; retrieval models; information theory; multimedia retrieval; hypertext; information network and inference. Prior to registering in this course, students should have a background in algorithms and data structures, and database and information retrieval comparable to the senior undergraduate level.

CS 825 Image Processing (3)
Image models; image transformations; enhancement and restoration techniques; image segmentation; feature extractions and higher level descriptions. Prior to registering in this course, students should have a background in image processing, and numerical and symbolic computing comparable to the senior undergraduate level.

CS 826 Bioinformatics and Biomedical Applications (3)
This course provides an introduction to research in bioinformatics, which is the analysis of biological and medical data. Topics include sequence and image analysis, modeling of complex processes, biomedical database organization, and biomedical data mining. Selected biomedical data applications are also examined.

CS 827 (327) Computer Audio (3)
Representation of audio, audio compression, spatialization and surround sound. Analysis and synthesis of sound waveforms. Speech and music. Temporal and spectral processing.

CS 828 (305) Human Computer Communication (3)
Theory and practice related to the design and implementation of usable software and easy-to-learn interfaces. Specific topics will include user-centered design and task analysis; prototyping and the iterative design cycle; interface design and methods of evaluation.

CS 829 Information Theory and Applications (3)
This course covers the fundamentals of information theory and its application in content

distribution over the Internet. Topics covered include: information theory, channel codes, content distribution network, and peer-to-peer network coding. Prior to registering in this course, students should have a background in Data Communications and Networks comparable to the senior undergraduate level.

CS 830 Machine Learning (3)
Models of learning; inductive inference; constructive and selective induction; learning from examples; explanation-based learning; machine discovery; grammatical inference; knowledge acquisition; applications. Prior to registering in this course, students should have a background in artificial intelligence comparable to the senior undergraduate level.

CS 831 Knowledge Discovery in Databases (3)
Knowledge discovery from databases is the nontrivial extraction of implicit, previously unknown, and potentially useful information from databases. This course focuses on data sources, extraction techniques, efficiency concerns, and measures of novelty and usefulness. Prior to registering in this course, students should have a background in database and information retrieval, and artificial intelligence comparable to the senior undergraduate level.

CS 833 Operating Systems (3)
Multiple processes and scheduling; resource management; storage management; file systems; deadlock problem; queuing models; distributed systems; fault tolerant systems; operating systems for parallel architectures. Prior to registering in this course, students should have a background in an introduction to operating systems, and computer system architecture comparable to the senior undergraduate level.

CS 834 Software Security (3)
This course surveys emerging research directions in software-based protection mechanisms, with an emphasis on approaches that employ programming language technologies to address security challenges. Topics covered include: Java security, stack inspection, inlined reference monitors, proof-carrying code, certifying compilers, type systems for information flow control, characterization of enforceable policies. Prior to registering in this course, students should have a background in operating systems and either programming languages or introductory compiler design comparable to the senior undergraduate level.

CS 835 Pattern Recognition (3)
Statistical pattern recognition; parameter estimation and supervised learning; nonparametric techniques; linear discriminant functions; unsupervised learning and clustering; syntactic pattern recognition; applications. Prior to registering in this course, students should have a background in algorithms and data structures comparable to the senior undergraduate level.

CS 836 Rough Sets and Applications (3)
Theory of rough sets is a fundamental mathematical methodology for modelling classification or decision problems involving imprecise or uncertain information. Its implications include pattern classification, data mining, machine learning, control algorithm acquisition from data, circuit design and others. The course will provide the basics of the methodology and will include the study of the above applications of rough sets. Prior to registering in this course, students should

have a background in discrete computational structures, artificial intelligence and statistical methods comparable to the senior undergraduate level.

CS 837 Information Visualization (3)
Information Visualization focuses on the design, development, and study of interactive visualization techniques for the analysis, comprehension, exploration, and explanation of large collections of abstract information. Topics to be covered include principles of visual perception, information data types, visual encodings of data, representations of relationships, interaction methods, and evaluation techniques.

CS 838 Uncertain Reasoning in AI (3)
Advances in using uncertain knowledge to make decisions rationally and effectively (for diagnosis, trouble shooting, robot navigation, etc.). Focus on probabilistic approach and graphical modeling to aid inference. Topics include criteria for uncertainty management, comparison of schemes, Bayesian/Markov networks, influence diagrams, chain graphs, inference algorithms, elicitation and learning of belief networks. Prior to registering in this course, students should have a background in discrete computational structures, artificial intelligence and statistical methods comparable to the senior undergraduate level.

CS 839 Web Intelligence and Electronic Commerce (3)
The course investigates research topics related to Web Intelligence and Electronic Commerce. The topics include: web technology, network infrastructure, web-based business models, agents, Extended Markup Language, web mining, security, web information filtering and retrieval, and intelligent information systems.

CS 872 Software Engineering (3)
Review of fundamental concepts; project planning; requirements analysis; program design, implementation and testing; object-oriented development; metrics and cost estimation; software reuse; CASE technology; configuration management; software engineering and Ada. Prior to registering in this course, students should have a background in software engineering methodology comparable to the senior undergraduate level.

CS 875 Database Systems (3)
Database management system architecture; relational, network and hierarchical data models; theoretical and practical aspects of database applications; study of data definition and data manipulation facilities and database management systems; security and integrity; distributed database management system architecture. Prior to registering in this course, students should have a background in advanced topics in database systems, and database and information retrieval comparable to the senior undergraduate level.

3. Attending One Co-op Information Session

The graduate co-op students are all required to attend one information session to get advises on how to choose courses, have an academic plan to fit the work terms, basic MSc co-op option requirements etc. which is provided by the graduate co-op committee chair.

4. Supervisor

All graduate co-op students should be assigned a supervisor. By the end of the first work term, each graduate co-op student must have a supervisor. The student should contact your supervisor to get advices on choosing the work term report topic. The supervisor is responsible for reading and evaluating the report.

5. Work term report

5.1 The format and the basic requirements

The standard of graduate co-op work term report should be higher than the undergraduate co-op work term report but lower than the project based report. It is a formal technical report. The overall format is similar to the undergraduate work term report. It should be about 25 pages long from introduction to conclusion.

The report should contain the following items:

- The cover page with report title, student ID, student name, and the program name. (ie Graduate Co-op Work Term Report)
- A letter of submittal. (a formal business letter)
- An executive summary. (one page long)
- A table of contents.
- Introduction, Background review, Methodologies, Analysis, Conclusion, References. (Future work and Appendix are optional.)

5.2 When should the student start writing the co-op report?

Students should start to think about the topic and have an idea what to write about by the end of the second month of CS602 which is the second work term. The student should contact the supervisor and discuss about the feasibility of the topic. By the end of the third month, the student should hand in a table of contents. By the end of the fourth month, the student should hand in a draft of introduction, background review, methodologies etc.

6. Presentation

The presentation requirements are similar to the seminar course. It is about 30 minutes long. The graduate co-op committee members and the supervisor should attend the presentation. The student should ensure that his/her supervisor approve the presentation material beforehand.

7. Transfer to Course-based or Project-based program

If the student fails to get the job placement and there is no opportunity to complete the co-op requirements, the student should transfer back to the course based program. However If the student consults with his/her supervisor, and the supervisor would like to continue supervise the student, the student can transfer to the project based or thesis based program and complete the requirements of the program.