CARLETON UNIVERSITY

Department of Systems and Computer Engineering

SYSC 5101 **Design of High Performance Software** **Winter 2012**

Course Handout

Instructor:

Dr. George Yee, Rm 4230ME, (613) 520-2600 X1962, gmyee@sce.carleton.ca

Course Description and Objectives:

Designing software to demanding performance specifications. Design analysis using models of computation, workload, and performance. Principles to govern design improvement for sequential, concurrent and parallel execution, based on resource architecture and quantitative analysis.

Performance Concepts: Performance quantities and how they are measured; resources in software and hardware.

Description of designs: Behaviour is described by *activity graphs*, based on Use Cases; structure is described by *module models*; connections are defined between these two and the special attributes of modules which are also concurrent threads or processes are added. Workload and parameters are discussed.

Analysis and prediction of performance: Performance metrics related to the models (descriptions) above; the mechanics of obtaining the metrics by setting up and solving models. The main models are Petri nets, queueing models, layered queueing for concurrent and parallel task systems, and task graphs. We will aim for an understanding of the semantics of the models, and what they are good for, but we will not go into the theories they are based on, or into the solvers. Emphasis on layered queueing, due to its special capabilities.

Architectural patterns: Throughout the above topics will be woven concepts of systems built of common architectural patterns, the effect of the patterns on performance (and how the effect depends on the parameters of the system). The use of the patterns to improve performance. Advanced patterns include replication, partitioning, fast paths, optimistic design. Patterns for reliability, and their effect on performance.

Project: An important part of the course will be a project to study a system of special interest to the student, or proposed by the instructor. Projects will involve using software modeling and analysis tools, which can be used in the lab or from home or (in some cases) from work. The project will take about 5 hours a week in the second half of the term.

References:

There are many books that describe performance modeling concepts; some that are widely available are:

- Edward D. Lazowska, John Zahorjan, G. Scott Graham, Kenneth C. Sevcik, Quantitative System Performance Computer System Analysis Using Queueing Network Models, Prentice Hall. (free download available from http://www.cs.washington.edu/homes/lazowska/qsp/)
- Daniel A. Menasce, Lawrence W. Dowdy, and Virgilio A.F. Almeida. Performance by Design: Computer Capacity Planning By Example Prentice Hall. ISBN-10: 0130906735.
- C. U. Smith has written widely on software performance.

- C. U. Smith and L. G Williams. Performance Solutions: A Practical Guide to Creating Responsive, Scalable Software. Addison Wesley 2002.

Course Outline:

1. Performance concepts and requirements.

2. Performance measurement. Workloads.

3. Performance models. Cures for performance problems.

4. Memory hierarchy effects. Queueing analysis.

5. Queueing Analysis cont.

6. Software resources.

7. Layered resource effects.

8. Measurement and tools.

9. Schedulability analysis for hard realtime systems.

10. Hard realtime systems cont.

11. Patterns for improving performance.

12. Project Presentations.

13. Review.

Grading Scheme:

- Two assignments on applying methods and tools, 20%
- A project with an oral presentation (10%) and a final report (30%), 40%
- A short written final exam on concepts and terminology. 40%

Important Notes:

1. The **final exam** is for evaluation purposes only and will not be returned to the student.
2. **Students with Disabilities** requiring academic accommodations in this course are encouraged to contact the Paul Menton Centre for Students with Disabilities (500 University Centre) to complete the necessary forms. After registering with the Centre, make an appointment to meet with me in order to discuss your needs.
3. **Plagiarism** and cheating at the graduate level are viewed as being particularly serious and the sanctions imposed are accordingly severe. Students are expected to familiarize themselves with and follow the Carleton University Student Academic Integrity Policy (See http://www1.carleton.ca/studentaffairs/academic-integrity/). The Policy is strictly enforced and is binding on all students. Plagiarism and cheating presenting another's ideas, arguments, words or images as your own, using unauthorized material, misrepresentation, fabricating or misrepresenting research data, unauthorized co-operation or collaboration or completing work for another student – weaken the quality of the graduate degree. Academic dishonesty in any form will not be tolerated. Students who infringe the Policy may be subject to one of several penalties including: expulsion; suspension from all studies at Carleton; suspension from full-time studies; and/or a reprimand; a refusal of permission to continue or to register in a specific degree program; academic probation; or a grade of Failure in the course.
4. Every student should have a copy of our **Health and Safety Manual**. An electronic version of the manual can be found at: http://www.sce.carleton.ca/courses/health-and-safety.pdf.