Carleton University

Department of Systems and Computer Engineering

SYSC 2001 ***Foundations of Computer Systems*** ***Fall 2011***

Course Outline

Instructor:

C. Schramm, ME4444, 613–520-2600 x2470, schramm@sce.carleton.ca

Course Description:

The student is introduced to the underlying components of the computer, how the components function, and how they are controlled by software through machine language programming. The concepts originated in the 1940's, yet have endured to the modern day. The course covers major computer enhancements that have been engineered to improve performance, where performance is meant to include execution speed. These performance improvements are often enabled by technology advances that allow increasing numbers of transistors to be included in components; however, the improvements presented in this course deal with modifications to how the components function rather than on transistor technology. Several processor families, including the Intel Pentium and Motorola PowerPC are used as examples. The relationship between high level programming languages and machine languages is also discussed.

The course will emphasize how a computer supports the execution of instructions and external interactions by presenting a roadmap to the engineering of performance in computers. Short sequences of assembly language instructions (code fragments) will be used to expose concepts and issues where relevant. The knowledge gained in this course is a valuable prerequisite for subsequent courses that emphasize application concerns in how to program a computer, the synthesis of programs to solve problems, and software development issues.

Course Objectives:

1. To give an understanding of how computers work as machines.
2. To provide an understanding of how computers have been engineered to improve performance.
3. To reinforce basic programming concepts learned in first year courses.
4. To provide essential pre-requisite knowledge for the SYSC 2003 and the SYSC 3601 courses.

Learning Expectations:

Students are required to attend class and the labs. Students are expected to read relevant sections of the text before they are discussed in class. The course material includes, but is not limited to, all indicated sections of the text. Lectures will cover highlights of the course material, but not necessarily all required sections. The lectures and lab may supplement the text with additional course material. Students are responsible for learning all of the course material.

Course Web Page: http://www.sce.carleton.ca/courses/sysc-2001

TA hours, lab locations and miscellaneous other important information will be posted on the web page. The web page is not guaranteed to be a complete record of what is announced and discussed in class, however. If you miss a class, you should ask a classmate what you missed.

Course Outline

Prerequisites:
Students who have not satisfied the prerequisites for this course must either
a) withdraw from the course, or
b) submit a prerequisite waiver online at: http://www.sce.carleton.ca/ughelp

Students who fail to clear the prerequisite requirement will be deregistered from the course after the last day to register for courses in the Fall term.

Textbook:
W. Stallings, "Computer Organization & Architecture", 8th Edition, Prentice Hall, 2010
Text Coverage: Chapters **1 – 5, 7, 9 – 12**
Note: earlier editions are acceptable.

Additional course notes and references may be distributed on the course web page to supplement the text.

Grading Scheme:
To pass the course, a student must pass the final examination (D- or better) and obtain an overall passing average (labs plus mid-term plus final exam). For students who pass the final exam, the final grade will be calculated as follows:

Labs:	20 %
Mid-term test:	25 %
Final exam:	55 %

Labs:
There will be five graded labs. Lab dates, pre-lab and materials will be made available on the course web page. Attendance at labs is mandatory and absolutely necessary to obtain a good grade.

The computer lab (MC6050/6055) is open whenever the building is open. You may use the lab at any time, except for those time slots when the lab is reserved for others.

Tutorial lab sessions are scheduled so that you may meet with the TAs for assistance.

Course Outline

Mid-Term Test:
The mid-term test (closed-book, no calculators permitted) will be held in the regular classroom at the scheduled class time sometime towards the middle of October.

Final Exam:
The final exam (closed book, no calculators permitted, Virgo reference sheet provided) will be held during the University's formal examination period in December.

The Final Examination is for evaluation purposes only and will not be returned to the student.

Students who miss the final exam may be granted permission to write a deferred examination (see the Undergraduate Calendar for regulations on deferred exams). These students have additional months to study and a less crowded examination schedule compared to those who write in December. As such, it is only fair to expect much better performance on the deferred examination than on the December final exam.

Students with Disabilities:
Students with disabilities requiring academic accommodations in this course must register with the Paul Menton Centre for Students with Disabilities for a formal evaluation of disability-related needs. Registered PMC students are required to contact the Centre, 613-520-6608, every term to ensure that I receive your Letter of Accommodation, no later than two weeks before the first assignment is due or the first in-class test/midterm requiring accommodations. If you require accommodation for your formally scheduled exam(s) in this course, please submit your request for accommodation to PMC by November 11th 2011 for Fall term (December exams).
.

Plagiarism
Plagiarism (copying and handing in for credit someone else's work) is a serious instructional offense that will not be tolerated. Please refer to the section on instructional offenses in the Undergraduate Calendar for additional information.

Course Outline

Course Schedule: Lecture-By-Lecture
NOTE: subject to change

Week 1 Ch. 1, Ch. 2
- brief history of computers, design for performance, Pentium family evolution
- computer functions
- introduction to Virgo, asm language programming

Week 2 Ch. 3 (plus - appendix 3A on timing diagrams)
- instruction fetch/execute, interrupts, I/O
- bus interconnection structures, simple timing diagrams
- assembly language programming cont.

Week 3 Ch. 10 (plus - Appendix 10B on endian and bit order)
- instruction set characteristics, operands
- types of operations
- endian schemes and bit ordering

Week 4 Ch 11 (Virgo)
- addressing modes
- instruction formats

Week 5 Thanksgiving – Midterm

Week 6 Ch. 5
- error correcting memory
- DRAM: synchronous, cache

Week 7 Ch. 7
- external devices, disks, I/O modules
- programmed I/O
- interrupt driven I/O
- DMA

Week 8 Ch. 4
- memory system characteristics, memory hierarchy, intro to cache memory
- cache memory design: size, mapping, replacement, write policy, number
- internal memory types, organization

Week 9 Ch. 9
- ALU, integer representation
- integer arithmetic

Week 10 finish Ch. 9
- IEEE floating point representation and arithmetic

Week 11 Ch 12
- processor organization, register organization, instruction cycle
- instruction pipelining

Week 12 finish Ch 12
- relationship of architecture and organization to high-level language
- control structures, built in data types
- functions/subroutines, parameter passing

Week 13 LAST CLASS - review