Linfield College, Computer Science Department
COMP 377: Computer Architecture
Calendar– spring 2016
Prof. Daniel Ford

COURSE DESCRIPTION:
COMP 377, computer organization and architecture is the study of how computers work from a software engineering perspective. We will investigate machine instructions, assembly language, representation and arithmetic, digital logic design, processors and data paths.

CREDITS: 3

PREREQUISITES: COMP 161 or instructor approval.

LECTURES: Renshaw 211
Section 1: TTh 9:55AM – 11:10AM

PROFESSOR: Daniel Ford
Office: Renshaw 210 Phone: x2706 E-mail: ford@linfield.edu
Office Hours: MWF 10:30—11:30 AM, TTh 1:30—2:30 PM, and by appointment.

COURSE OBJECTIVES and LEARNING OUTCOMES:
The goal of this course is to understand how computers translate programs into binary sequences and how circuit diagrams can be built that can execute these binary sequences. Students will create a 4-bit calculator/computer from digital logic gates and have it run simple programs. Students should be able to write recursive, and other similarly complex, assembly language programs and translate them into machine language.

RECOMMENDED RESOURCES:

- Exploring Digital Logic with Logisim by George Self, http://www.lulu.com/us/en/shop/george-self/exploring-digital-logic-with-logisim-ebook/ebook/product-21118223.html.
- Introduction to MIPS Assembly Language Programming by Charles W. Kann, http://chuckkann.com/books/IntroductionToMIPSAssembly.
- Digital Design and Computer Architecture by Harris and Harris from Morgan Kauffman
- Digital Fundamentals by Thomas L. Floyd from Pearson.
- Computer Organization and Design by Patterson and Hennessy from Morgan Kauffman

GRADES:
Grading for this course will be based on:

Exams	~36%
Quizzes	~14%
Homework/Assignments	~50%

Make-up exams will not be granted without prior arrangement or for reasons stated in the college catalog.

There are two midterms scheduled to cover digital logic design and one final centered on issues concerned with programming in and translating to and from assembly language.

ILLNESS/MEDICAL POLICY:
In the unfortunate event that you must miss class you are encouraged to contact your workshop partner to find out what we worked on in class and to pick up any materials that were handed out. In general, homework extensions and make-up exams will be granted as needed on an individual basis.

DISABILITY SUPPORT:
Students with disabilities are protected by the Americans with Disabilities Act and Section 504 of the Rehabilitation Act. If you are a student with a disability and feel you may require academic accommodations please contact Learning Support Services (LSS), as early as possible to request accommodation for your disability. The timeliness of your request will allow LSS to promptly arrange the details of your support. LSS is located in Melrose Hall 020 (503-883-2562). We also encourage students to communicate with faculty about their accommodations.

CLASS/GROUP PARTICIPATION:

Attendance is required. Students may be asked to work in front of the class on the white boards.

Computer science requires tolerance, individual contributions, teamwork and the ability to learn from others. For the academic endeavor to succeed, we must treat each other with civility, courtesy and respect. You will frequently be encouraged to work in. pairs. Every student will be expected to make pertinent and substantive contributions to every group they are a member of.

One learns more when working in groups when the others help them to remain productive: to generate better ideas and evaluate them faster and to spend less time on technical difficulties with little pedagogical value. If, however, the group is doing the work for you, supplying you with ideas and evaluating and choosing which ideas to use to solve your problems, then they are doing you a disservice by reducing your opportunity to generate and evaluate your own ideas and become more proficient.

ACADEMIC CONDUCT:

In this course we will adhere to the college policy on academic honesty, as published in the Linfield College Course Catalog. Please review this policy if you are not already familiar with it.

Linfield College operates under the assumption that all students are honest and ethical in the way they conduct their personal and scholastic lives. Academic work is evaluated on the assumption that the work presented is the student's own, unless designated otherwise. Anything less is unacceptable and is considered a violation of academic integrity. Furthermore, a breach of academic integrity will have concrete consequences that may include failing a particular course or even dismissal from the college.

Violations of academic integrity include but are not limited to the following:

- *Cheating*: Using or attempting to use unauthorized sources, materials, information, or study aids in any submitted academic work.
- *Plagiarism:* Submission of academic work that includes material copied or paraphrased from published or unpublished sources without proper documentation. This includes self-plagiarism, the submission of work created by the student for another class unless he or she receives consent from both instructors.
- *Fabrication*: Deliberate falsification or invention of any information, data, or citation in academic work.
- *Facilitating Academic Dishonesty*: Knowingly helping or attempting to help another to violate the college's policy on academic integrity.

Note that plagiarism is claiming that someone else’s work or ideas are your own. Copying or using someone else’s work or ideas is not plagiarism unless you claim that the work or ideas are your own. By submitting academic work you are implicitly claiming, unless you explicitly state otherwise, that the submitted work and the ideas presented in it are your own.

In this class you will earn credit by combining and adapting basic building blocks, e.g. common data structures and algorithms to solve a particular problem or accomplish a specific task. You are expected to generate, evaluate, and illustrate your own ideas on how these building blocks can be put together to solve particular problems. In order to avoid plagiarizing,

- Obey the 30 second long-term memory rule. We all get stuck now and again. If you get help on a homework problem and/or glance at someone else’s work, don’t just implement their solution. Instead take a minute to think about their idea why it works or doesn’t work and how you might improve or customize it before writing your own solution.
- Always add comments to explain any non-trivial solution you submit, and never submit anything that uses a technique or solution that you don’t completely understand.