Spring 2017

SOUTHERN MAINE COMMUNITY COLLEGE
envision a future

South Portland, Maine 04106

Department of Computer and Information Sciences

Title: Computer Organization	**Catalog Number: CSCI 250**
Credit Hours: 4	**Total Contact Hours: 60**
Lecture (or Lab): Lecture	**Instructor: Dr. Anne Applin**
Office Hours – Location: CSEC 025	**Contact Information:** Office phone: 207-741-5778
MTWR: 1:00 – 2:00	aapplin@smccme.edu
	Google Voice: 207-200-5853 (text or voice)

Other hours available by appointment

Course Syllabus

Course Description

This course introduces the hardware components of a computer focusing on their function, design, and performance characteristics including discussions of recent developments. The hardware-to-software interface will be examined where high-level language statements are reduced to a sequence of assembly language statements, each of which represents a single primitive instruction of the processor. Students will learn binary and hexadecimal numbers and how variables, arithmetic/logic operations, branching, loops, arrays, and function calls are written in the "native language" of a microprocessor (in this case the MIPS R3000 32-bit RISC instruction set). Advanced topics include performance features such as processor caches and pipelining. Most of these topics are designed to enhance your problem-solving and logical reasoning abilities. **Prerequisite:** Successful completion of CSCI160 Object Oriented Design and Programming.

Course Objectives

After successfully completing the course, the student will be able to:

1. understand the interaction of devices, busses, and processors
2. demonstrate the ability to do arithmetic with binary, decimal and hexadecimal number systems
3. demonstrate the ability to do arithmetic with IEEE floating point numbers in binary
4. create assembly language programs using the SPIM MIPS R3000 Emulator
5. create high level programming constructs in assembly language including if-then-else , loops, function calls, and recursion
6. demonstrate an understanding of the addressing for one and two dimensional arrays in assembly language
7. compute computer performance given clock cycles and cycles per instruction
8. calculate performance enhancements using caches and pipelining.

Learning Outcomes

This course seeks to develop the following core abilities:

Global Awareness / Diversity –

- Identify resources and strategies needed to problem solve and/or achieve goals inclusive of diverse perspectives and experiences.
- Demonstrate the ability to work collaboratively with people from diverse backgrounds in pursuit of a common objective or goal utilizing interpersonal skills that are essential to team building, conflict resolution and cross-cultural communication.

Critical Thinking –

- A student can interpret information logically by selecting and organizing relevant facts and opinions and identifying the relationships among them.
- A student can analyze an issue or problem by separating it into its component parts and investigating the relationship of the parts to the whole.
- A student can synthesize information by combining ideas from multiple sources to come to an independent conclusion.
- A student can evaluate information by making informed judgments as to whether the information is accurate, reliable or useful.
- A student can apply theory to practice.

Communications –

- Demonstrate a command of the English language
- Identify and extract relevant data from written and oral presentations

Quantitative Methods –

- Recognize problems that can be solved with quantitative methods
- Identify the quantitative components of a problem
- Select and appropriate mathematical method to solve a problem
- Demonstrate accurate computational and/or algebraic skills to solve a problem
- Estimate the reasonableness of answers to problems

Google Voice Contact: When texting or leaving voice mail on the Google contact number, please identify yourself first. I will see your number but no name so I need to know who I'm talking to.

Learning Objectives

Week 1 *Chapter 1:*
to describe the big picture of how a computer turns code into instructions
to describe the interactions of the various hardware components in the box
to describe program performance and optimization
to calculate processing latencies and relative latencies

Week 2 to define the units of time, storage, and throughput
to describe the effect of bandwidth on throughput
Section 2.4
to describe data representations and computer arithmetic
to demonstrate the ability to convert from hexadecimal to binary and back
to demonstrate the ability to do binary arithmetic
to describe overflow and underflow
to calculate the limits of computation in *n* bits

Week 3 *Section 2.4*
to demonstrate the ability to do arithmetic with positive and negative binary numbers
to demonstrate the ability to convert to two's complement for calculation
Section 3.5
to describe floating point representations
to demonstrate the ability to do addition, subtraction and multiplication with floating point binary numbers.
to describe character encoding
to describe error checking schemes

Week 4 **Test 1**
to describe the concept of instruction sets used in hardware
to demonstrate how a computer uses the instruction set to do computation
to describe the relationship between assembly language and the instruction set
to be able to demonstrate the ability to write a simple assembly program to manipulate data using logical operations

Week 5 *Section 2.6*
to create MIPS assembly programs to do shifting of bits in the register
to be able to determine the sign of a number by examining the high order bit using MIPS assembly language
to be able to write a MIPS assembly program to perform simple arithmetic
Programming Project 1

Week 6 *Section 2.7*
to create MIPS assembly language programs that include branching
to create MIPS assembly language programs that use decision structures
Programming Project 2

Week 7 to describe 16 and 32 bit memory addressing
to describe array addressing
to demonstrate the ability to calculate the address of any given element in an array

Week 8 to demonstrate how arrays are created in MIPS assembly language
to create a MIPS assembly language program to perform array manipulation
Programming Project 3

Week 9 **Test 2**
to demonstrate how to implement a switch statement in MIPS assembly language
Section 2.8
to demonstrate how function calls are implemented using stacks in MIPS assembly language
to describe reentrant code in MIPS

Week 10 to create MIPS assembly language programs that use recursive functions
Section 2.5
to demonstrate the ability to decode machine language instructions into assembly language instructions
to describe the two pass assembler

Week 11 to describe performance indicators
to compute speedup & execution times based on clock cycles of execution and clock rate
to be able to calculate average cycles per instruction

Week 12 to describe memory hierarchies
to be able to calculate the hit ratio
to describe write through versus write back cache memory
Week 13 to describe pipelining
to be able to calculate throughput and latency of pipelined systems
Week 14

Final Exam (comprehensive)

Exam Policies: You will be allowed to create and use a one page assistance sheet during exams including the final. The single 8.5 X 11 inch piece of paper can have anything on front and back that you wish to have handy during exams. You may not give your assistance sheet to anyone else. Each student who wishes to use one must create her own.

Course Requirements

Students will complete 5 homework assignments. In addition to coding dozens of small example MIPS programs as a class, students will individually create 5 problem solution designs and code those solutions, take 3 tests during scheduled class times and complete a comprehensive final examination.

Student Evaluation and Grading

Two in-class tests	45%
Programs, Homework, Quizzes	30%
Final Exam (Comprehensive)	25%

Grading Scale:

93 – 100	A
90 - 92.99	A-
87 - 89.99	B+
83 - 86.99	B
80 - 82.99	B-
77 - 79.99	C+
73 - 76.99	C

70 - 72.99	C-
67 - 69.99	D+
63 - 66.99	D
0 - 62.99	F

Text, Tools and / or Supplies

Go to zybooks.com and enter book code**:** SMCCMECSCI250ApplinFall2017. The cost is $68 which is refundable if you drop the course. You should also have a notebook for taking notes and a writing instrument. It is strongly recommended that the student have a USB drive to store backup copies of all programming assignments. Students should be prepared to work out side of class 6- 8 hours per week on homeworks, projects and preparation for class. It is possible to get a pdf (for free) of an older version of the text. You may opt to do this.

End-of-Course Evaluation

In order to gain access to final course grades, students must complete evaluations for each course attended at SMCC. Evaluations are submitted online and can be accessed through the student portal site. Students can access the course evaluation report beginning two weeks before the end of classes. The deadline for submission of evaluations occurs 24 hours after the last day of classes each semester. Instructors will announce when the online course evaluation is available.

ADA (Americans with Disabilities Act):

Southern Maine Community College is an equal opportunity/affirmative action institution and employer. For more information, please call 207-741-5798.

If you have a disabling condition and wish to request accommodations in order to have reasonable access to the programs and services offered by SMCC, you must register with the Disability Services Coordinator, Sandra Lynham, who can be reached at 741-5923.

Further information about services for students with disabilities and the accommodation process is available upon request at this number. Course policies about online testing are modified to suit each individual's accommodations.

SMCC Pay-for-Print Policy

In an effort to control the escalating cost of supplies and to encourage students to conserve resources, SMCC charges for printing. Students receive a $20 credit every semester. This credit resets for each semester and extra credit is not rolled over to the next semester. Per page costs are as follows:

- 8.5"x11" black and white: $0.10 per page
- 8.5"x11" color: $0.50 per page
- 8.5"x14" and 11"x17" black and white: $0.20 per page
- 8.5"x14" and 11"x17" color: $1.00 per page

Duplex (two-sided) pages are discounted 50% from the listed page costs.

Students can monitor their remaining credit and number of pages printed by visiting the IT Help tab on MySMCC or by checking the Printing Information icon in the lower right corner of the screen while logged in to an SMCC computer.

More information about the Pay-for-Print Policy is available on the IT Help tab on MySMCC.

Add-Drop Policy

Students who drop a course during the one-week “add/drop” period in the fall and spring semesters and the first three days of summer sessions receive a 100% refund of the tuition and associated fees for that course. Please note any course that meets for less than the traditional semester length, i.e., 15 weeks, has a pro-rated add/drop period. There is no refund for non-attendance.

Withdrawal Policy

A student may withdraw from a course only during the semester in which s/he is registered for that course. The withdrawal period is the second through twelfth week of the fall and spring semesters and the second through ninth week of twelve-week summer courses. This period is pro-rated for shorter-length courses. To withdraw from a course, a student must complete and submit the appropriate course withdrawal form, available at the Enrollment Service Center (no phone calls, please). The designation “W” will appear on the transcript after a student has officially withdrawn. A course withdrawal is an uncompleted course and may adversely affect financial aid eligibility. Failure to attend or ceasing to attend class does not constitute withdrawal from the course. There is no refund associated with a withdrawal.

Plagiarism Statement

Adherence to ethical academic standards is obligatory. Cheating is a serious offense, whether it consists of taking credit for work done by another person or doing work for which another person will receive credit. Taking and using the ideas or writings of another person without clearly and fully crediting the source is plagiarism and violates the academic code as well as the Student Code of Conduct. If it is suspected that a student in any course in which s/he is enrolled has knowingly committed such a violation, the faculty member should refer the matter to the College’s Disciplinary Officer and appropriate action will be taken under the Student Code of Conduct. Sanctions may include suspension from the course and a failing grade in the course. Students have the right to appeal these actions to the Disciplinary Committee under the terms outlined in the Student Code of Conduct.

CSCI 250–- Collaboration Policy

CLASSWORK / HOMEWORK

You may collaborate on CLASS WORK ASSIGNMENTS in and out of class. However, your final answers MUST be YOUR OWN. This means that you MAY work together to solve the problems, but the final answers must be done INDEPENDENTLY. (You may NOT copy another person's work!)

QUIZZES and TASKS

No discussion of any kind with anyone but the instructor is allowed. Use of unauthorized written material, cell phones, or other messaging tools is not allowed.

PROJECTS and OTHER ASSIGNMENTS

Discussion of techniques in a natural language (such as English) is allowed. Discussion of an assignment in a computer or algorithmic language (such as Java) is NOT allowed. Strictly avoid sharing or exchanging literal statements of computer code or program files. Computer language questions are to be limited to the language and should not concern the assignment. WHEN IN DOUBT, SEE THE INSTRUCTOR! Stealing, giving or receiving passwords, code, designs, drawings, diagrams and/or text from ANY other person (whether from on-campus or off-campus) is NOT allowed. Every line of code that you turn in must be your own!

Any of the following also constitutes cheating:

1. Having a copy of a program that is not your own.
2. Accessing or viewing anyone else's work.
3. Giving anyone else access to your work.
4. Any attempt to collaborate on projects.
5. Any attempt to deceive the instructor.

Student responsibilities include:

1. Secure disposal of code and report of missing printouts.
2. Avoidance of other students who act unethically.
3. Keeping your program solutions to yourself.

The Penalty

Violations of the collaboration policy will result in a zero on the assignment in questions and will be referred to the Disciplinary Committee for further action**.**

I have read, and understand the syllabus and collaboration policy for CSCI 250.

PRINT Name (last, first, mi): ____

Signature: _____________________________________