Course Syllabus Spring 2017

Course Information
CS/SE/TE 3340 Computer Architecture -- Section: 501
Time: MW 5:30 – 6:45 pm
Location: SLC 2.303

Professor Contact Information
Dr. Karen Mazidi
Email: Karen.Mazidi@utdallas.edu
Office: ECSS 3.203
Office phone: 972-883-3868
Office hours: Monday - Thursday 4:00-5:00 pm or by appointment

Mentoring Center Information
ECSS 4.415 Main Room / Walk-in Tutoring
ECSS 3.620 and 3.621 Review/Reword Rooms
login at: https://csmc.utdallas.edu/login.php to see hours of operation and other information

TA Information
TA assignments are usually made by the second week of class. The TA name and email will be added to CourseBook automatically. An announcement will also be posted in eLearning.

Course Pre-requisites, Co-requisites, and/or Other Restrictions
CE 1337 or CS 1337 or TE 1337 with a grade of C or better
and
CE 2305 or CS 2305 or TE 2305 with a grade of C or better

Course Description
This course introduces the concepts of computer architecture by going through multiple levels of abstraction, and the numbering systems and their basic computations. It focuses on the instruction-set architecture of the MIPS machine, including MIPS assembly programming, translation between MIPS and C, and between MIPS and machine code. General topics include performance calculation, processor datapath, pipelining, and memory hierarchy. Credit cannot be received for both courses, (CS 3340 or SE 3340 or TE 3340) and (CE 4304 or EE 4304).

Student Learning Objectives/Outcomes
After successful completion of this course, students will:

1. Be able to write a fully functional, stand-alone medium size assembly language program
2. Have an ability to represent numbers in and convert between decimal, binary, and hexadecimal and perform calculations using 2's complement arithmetic
3. Understand the basic model of a computer including the datapath, control, memory, and I/O components
4. Be able to program efficiently in an assembly level instruction set, including the use of addressing modes and data types
5. Understand the role of compilers, assemblers, and linkers and how programs are translated into machine language and executed

6. Be able to demonstrate comprehension of a pipelined architectures including datapaths and hazards
7. Be able to demonstrate comprehension of computer performance measures and their estimation
8. Understand the memory hierarchy including caches and virtual memory

---

Required Textbooks and Materials

Computer Organization and Design, Fifth Edition: The Hardware/Software Interface (The Morgan Kaufmann Series in Computer Architecture and Design) Patterson and Hennessy, ISBN 9780124077263.

There are many editions of this book, make sure you get the 5th edition with MIPS.

> This course will involve coding in the MIPS assembly language using MARS (MIPS Assembler and Runtime Simulator). Get it free here:
> http://courses.missouristate.edu/KenVollmar/mars/index.htm
> The home page of MARS also has links for tutorials. See the Download link for download instructions. This should work on any computer with Java installed.

Suggested Course Materials

Any additional materials will be available from the course webpage.

Academic Calendar

Important Dates

- Semester begins Monday January 9
- Martin Luther King Day Monday January 16 (School Closed)
- Census Day January 25
- Mid-term exam Week 7 (Week of February 20)
  - check eLearning for announcement
- Spring break March 13 – 18 (no classes)
- Last week of class – week of April 24
- Final exam week – week of May 1 – May 8
  - check CourseBook after Census Day

Topics and Sequence (a *tentative* schedule)

- Week 1 – **A Peek Under the Hood** – Read 1.1-1.5, 2.4, 2.9, 2.12
  - Course introduction
  - 8 Great ideas in computer architecture
  - Compiling/Linking to machine code
  - Installing MARS
  - Numbering and coding systems
- Week 2 – **MIPS Intro** – Read 2.1 – 2.3, 2.5, Appendix A
  - Getting started with MIPS
  - MIPS operations and operands
  - Load and store instructions
- Week 3 – **Arithmetic and Logic Instructions** – Read 2.6, 2.10, Appendix A
  - Arithmetic instructions
  - Logical instructions

- Addressing modes
- MIPS syscalls

- Week 4 – **Control Structures** – Read 2.7 – 2.8
  - Branching for decisions
  - Branching for loops
  - Subroutines
- Week 5 – **Performance** – Read 2.13 - 2.14, 1.6 – 1.10
  - C sort example
  - Arrays v. Pointers
  - Processing character arrays
  - Performance measures
- Week 6 – **Floating Point Numbers** – Read 3.1 – 3.5
  - MIPS arithmetic operations
  - Floating Point representation
  - Full MIPS instruction set and pseudo instructions
- Week 7 – Exam 1
  - Review for MidTerm
  - MidTerm
- Week 8 – **Other Architectures** – Read 2.16 – 2.18, 3.6 – 3.7
  - Review Exam
  - Intel x86, ARM architectures
- Week 9 – **Compilers and Logic** -- Read 2.15 (in eLearning) and Appendix B.1 – B.7
  - Compiler optimization
  - Logic Design
- Spring Break
- Week 10 – **Inside the CPU** – Read 4.1 – 4.9
  - MIPS non-pipelined CPU
  - MIPS pipelined CPU
- Week 11 – **CPU Optimization** – Read 4.10 – 4.15, Appendix B.7 – B.11
  - Optimization techniques
  - Memory logic
- Week 12 – **Memory Hierarchy** – Read 5.1 – 5.9
  - Memory technologies
  - Memory hierarchy
  - Virtual memory
- Week 13 – **Advanced Memory Concepts** – Read 5.10 – 5.16
  - Advanced memory management
- Week 14 – **Parallel Programming** – Read through Chapter 6
  - Many types of parallelism
  - GPUs
  - Parallel Programming
- Week 15
  - Review
- Exam

Grading Policy

- Assignments (30% of the course grade): There will be several programming/homework assignments. You will have at least a 1-week time frame for completion.
- Mid-term exam (30% of the course grade).
- Final exam (30% of the course grade).
- Quizzes and class attendance/participation (10%).

Letter grades will be assigned as follows:
Below 60 is an F

100 is an A+
Otherwise:
Grades ending in 7,8,9 are +
Grades ending in 0,1,2 are –
Grades ending in 3,4,5,6 are neither + nor -

Course & Instructor Policies

- Computers and phones must remain out of sight during class. Students should take notes by hand using either physical or electronic paper. Here is why: http://www.scientificamerican.com/article/a-learning-secret-don-t-take-notes-with-a-laptop/
- Assignments must be turned in on the due date, by midnight.
- Late assignments are deducted by 10% on the first and second days late. After two days, the assignment will not be accepted.
- Makeup exams are not given unless prior permission has been granted due to extenuating circumstances. Do not ask to take your final exam early.
- Do not turn in programming code or other work that is not your own. This will result in a zero for all parties. Discussion and collaboration are good things, turning in someone else's work as your own is not.
- **If you do not agree with a grade you have been given, you must make your case within a week of receiving the grade or the grade stands as is.**

Comet Creed

This creed was voted on by the UT Dallas student body in 2014. It is a standard that Comets choose to live by and encourage others to do the same:

"As a Comet, I pledge honesty, integrity, and service in all that I do."

Attendance Policy

Per Computer Science administration guidelines, please be aware that ***3 consecutive absences leads to one letter grade drop. Four consecutive absences lead to an F.***

DO NOT ask to take your final exam early. We will not know the exact date of the final exam until after Census Day. At that time it will be posted automatically for all of your courses in CourseBook. Final exams are scheduled for Tues May 2 – Mon May 8. Therefore it would be a really bad idea to book a flight before Tuesday May 9.

UT Dallas Syllabus Policies and Procedures

The information contained in the following link constitutes the University's policies and procedures segment of the course syllabus.

Please go to http://go.utdallas.edu/syllabus-policies for these policies.

The descriptions and timelines contained in this syllabus are subject to change at the discretion of the Professor.