King Fahd University of Petroleum and Minerals
College of Computer Sciences and Engineering
Department of Computer Engineering

COE 301 Computer Organization (3-3-4)
ICS 233 Computer Architecture & Assembly Language (3-3-4)

Instructor: Dr. Marwan Abu-Amara
Office: 22-145
Phone: 1632
E-mail: marwan@kfupm.edu.sa

Term: 171 (1st term 2017–2018)
Day & Time: UTR 08:00 AM – 08:50 AM
Location: 22-132
Prerequisite: COE 202 and ICS 201
Textbook: David A. Patterson and John L. Hennessy, *Computer Organization and Design: The Hardware /Software Interface*, Fifth Edition, Morgan Kauffmann Publishers, 2013.
Office Hours: UTR 10:00 AM – 10:50 AM (or by appointment)
Web Site: http://faculty.kfupm.edu.sa/COE/marwan

Tentative Grading Policy:

- Prog. Assignments **10%** (Each prog. assignment may carry a different weight)
- Quizzes **10%** (Each quiz may carry a different weight)
- Laboratory **10%**
- Project **15%**
- Major Exam I **15%** (Saturday October 21, 2017 – 10:00 AM)
- Major Exam II **20%** (Saturday December 02, 2017 – 10:00 AM)
- Final Exam **20%** (Tuesday January 02, 2018 – 07:00 PM)

Course Learning Outcomes

1. Describe the organization and operation of integer and floating-point arithmetic units. **(COE 301)**
2. Apply the knowledge of mathematics to processor performance analysis. **(COE 301)**
3. Design the datapath and control of a processor. **(COE 301)**
4. Describe the memory hierarchy and caches. **(COE 301)**
1. Analyze, write, and test MIPS assembly language programs. **(ICS 233)**
2. Use software tools for assembly language programming and for CPU design and simulation. **(ICS 233)**

IMPORTANT NOTES:

- All KFUPM regulations and standards will be enforced. Attendance will be checked each class. The KFUPM rule pertaining to a DN grade will be strictly enforced (i.e. **> 9 absences** will result in a DN grade).
- Only university approved/certified excuses will be accepted, and should be presented **no later than 1 week** after absence.
- Use of cell phones, smart phones, and tablets during class period and during exams is absolutely **prohibited**.
- Programming assignments are to be submitted **in class** on the due date during the class period. Late assignments will **NOT be accepted**.
- You have up to the next class period to object to the grade of a programming assignment, a quiz, or a major exam from the end of the class time in which the graded papers have been distributed back. If for some reason you cannot contact me within this period, send me an email requesting an appointment. The email should be sent before the next class period.
- **NO make-up exams**. ALL programming assignments and quizzes will be counted towards your grade.
- All exams are **common**.

Course Topics

1. ***Introduction***
   Introduction to computer architecture, assembly and machine languages, components of a computer system, memory hierarchy, instruction execution cycle, chip manufacturing process, technology trends, programmer's view of a computer system.

2. ***Review of Data Representation***
   Binary and hexadecimal numbers, signed integers, binary and hexadecimal addition and subtraction, carry and overflow, characters and ASCII table.

3. ***Instruction Set Architecture***
   Instruction set design, RISC design principles, MIPS instructions and formats, registers, arithmetic instructions, bit manipulation, load and store instructions, byte ordering, jump and conditional branch instructions, addressing modes, pseudo instructions.

4. ***MIPS Assembly Language Programming***
   Assembly language tools, program template, directives, text, data, and stack segments, defining data, arrays, and strings, array indexing and traversal, translating expressions, if else statements, loops, indirect jump and jump table, console input and output.

5. ***Procedures and the Runtime Stack***
   Runtime stack and its applications, defining a procedure, procedure calls and return address, nested procedure calls, passing arguments in registers and on the stack, stack frames, value and reference parameters, saving and restoring registers, local variables on the stack.

6. ***Interrupts***
   Software exceptions, syscall instruction, hardware interrupts, interrupt processing and handler, MIPS coprocessor 0.

7. ***Integer Arithmetic and ALU design***
   Hardware adders, barrel shifter, multifunction ALU design, integer multiplication, shift add multiplication hardware, Shift-subtract division algorithm and hardware, MIPS integer multiply and divide instructions, HI and LO registers.

8. ***Floating-point arithmetic***
   Floating-point representation, IEEE 754 standard, FP addition and multiplication, rounding, MIPS floating-point coprocessor and instructions.

9. ***CPU Performance***
   CPU performance and metrics, CPI and performance equation, MIPS, Amdahl's law.

10. ***Single-Cycle Datapath and Control Design***
    Designing a processor, register transfer, datapath components, register file design, clocking methodology, control signals, implementing the control unit, estimating longest delay.

11. ***Pipelined Datapath and Control***
    Pipelining concepts, timing and performance, 5-stage MIPS pipeline, pipelined datapath and control, pipeline hazards, data hazards and forwarding, control hazards, branch prediction.

12. ***Memory System Design***
    Memory hierarchy, SRAM, DRAM, pipelined and interleaved memory, cache memory and locality of reference, cache memory organization, write policy, write buffer, cache replacement, cache performance, two-level cache memory.