Program Software Engineering

Syllabus of Course 10145

Computer Organization and Assembly

Academic Year	**2025**
No. of course hours	**6.00 Semestrial hours [Lecture 4.00 + tutorial -2.00]**
Academic credits	**5.00**
Prerequisites	**Pre: 10006 Introduction To Computer Science -or- 10016 Introduction To Computer Science**
Please note that **The prerequisites are for all programs, you are required to be updated on the prerequisites you need according to your personal program.**	
Class Attendance	**Not mandatory**
Objectives	**- To implement the principles of logic system hardware.** **- To understand the basic structure of a digital computer and to study in detail the basic CPU unit organization in different ways of implementation till Modern computer technology.** **- To learn the basic of assembly language programing focused on the Hardware software interface.**
Abstract	**This course will give you an in-depth understanding of the inner-workings of modern digital computer systems in all the levels of abstraction and tradeoffs present at the hardware-software interface.** **The MIPS architecture is used as guiding example throughout the course.** **Topics include:** **- Performance of a processor system.**

	- Computer arithmetic (Binary integer and floating point number systems) - Boolean Algebra and logic gates. - Assembly language including instruction formats, addressing modes, instruction types. flow of control, the assembly process, linking, loading, Pseudo instructions, and Rules for Procedures. - Combinational logic and synchronous sequential logic, building the basic CPU units. - The processor Single cycle, Multi Cycle and pipeline implementation. - pipeline hazards. - interrupts and exceptions. - memory hierarchies.

Academic learning outcomes

Learning outcomes related to the content of the course	- The student could explain how modern computer works in all the levels of abstraction - The student could interpretant assembly instruction into machine instruction and transform it into a logic circuit - Vice versa The student could implement logic circuit and transform it into a machine instruction and interpretant assembly instruction - The graduate student could understand how to enhance performance via multicycle and parallelism (pipeline) and solve problems (Hazards) that parallel configuration make. - The graduate student can write and read a basic assembly code with regard to the hardware-software interface, include load store mechanism and procedure rules.

Learning outcomes - Skills

Problem solving: defining problems and identifying strategies. The ability to identify one or more approaches to problem solving without application in a specific context.
Programming: writing a working, high-quality program, adding to an existing program, reading, and understanding a program. The ability to identify types of variables, output and input, using conditionals, loops, functions, and the ability to read a simple program.
Integrative learning: linking to practical experience. The ability to identify connections between experience and similarly perceived ideas.

Further points of emphasis	**4 hours**

Lecture topics by weeks

The order of the topics can be changed at the lecturer's discretion.

1	**An introduction to the course. The basic concepts of a von Neuman architecture system.** **The importance of Instruction sets Architecture (ISA) RISC vs. CISC.** **- Digital systems and binary numbers.**
2	**Performance: Assessing and understanding Performance.** **Boolean Algebra and logic gates.**
3	**MIPS Assembly language: (R format I Format J Format), Arithmetic Logic operations,** **Data transfer, Conditional Branch, Unconditional JUMP. Addressing modes.**
4	**Combinational logic: building ALU and other arithmetic units** **synchronous sequential logic: building the register file**
5	**Single Cycle CPU Datapath and Control implementation**
6	**Advanced Single Cycle CPU critical path.** **Multicycle implementation (Finite state machine)**
7	**Multicycle implementation**

	Compiler, linker, loader. Rules for Procedures. (Stack and variables)
8	**Introduction to pipeline** **Pipeline: Data Hazard**
9	**midterm exam** **Continue advanced Data Hazard.**
10	**Pipelinine hazards: Control hazards. Branch prediction, pipeline scheduling** **Exceptions: The Interrupt Mechanism**
11	**floating point numbers (IEEE754)** **introduction to Memory Hierarchies**
12	**Memory Hierarchies**
13	**Measuring and improving Cache performance Virtual Memory & Review**

Tutorials / Labs topics by weeks

The order of the topics can be changed at the lab instructor's / tutor's discretion.

1	**Numbers representation (sign unsigned ,operation, overflow, radix)**
2	**Exercise: Performance and Boolean Algebra and logic gates.**
3	**Introduction to MARS and The MIPS ISA format**
4	**Lab1: Arithmetic and Logic and shift Instructions; implement simple for loop.**
5	**Lab2-3: I/O- Exception Handler- syscall**
6	**Lab4: Nested loops**
7	**Lab 5 :Arrays string and pointers (memory load and store mechanism)**
8	**Finish Lab5 and simulation for midterm exam**
9	**Lab6: Stack ,procedures (The conventions of variables)**
10	**final program assignment on nested procedure**
11	**Lab7:Floating point arithmetic on MIPS**
12	**Exercise: Cache**
13	**Review and advanced exercise on cache**

Course coordinator	**Dr. Busani Ram**
Language of instruction	**Hebrew**
Subjects for self-tutoring	
Textbooks and Recommended Bibliography	**1. Patterson and Hennessy, "Computer Organization & Design: The Hardware/Software Interface" 6th edition, Morgan Kaufmann Publishers, 2021.** **2. M.Morris.Mano and Michael d. Ciletti "DIGITAL DESIGN" 6th edition Pearson Education. Inc, 2017**

Course Requirements and Calculation of Final Grade

Task Type	Percentage of Final Grade
Final Exam Grade	80
Midterm Exam Grade	10
Homework Assignments	10
A project in a course where there is no Final Exam	0
A project in a course where there is a Final Exam	0
Final Grade	0

Clarification to pass the course:

In order to pass the course, students must fulfill the following conditions [excluding the English Beginners Course, Labs and Workshops]:

1. Final course grade of at least 60 [taking into consideration all the above course requirements].
2. Attendance according to the attendance requirement [see section regarding attendance].

Exam and Midterm Exam

Type of Midterm Exam	**Frontal (in the classroom)**
Duration of Midterm Exam	**70 minutes**
Location of Midterm exam	**Computer lab**
Duration of Final Exam	**210 minutes**
Location of Final exam	**Regular class (no computers)**
Permitted Material/Tools for Exams	**Any written or printed material**
Details of permitted materials for exam	
Formula Sheets	
Number of single-sided sheets	