Fayoum University **Faculty of computers & Information**

Computer Architecture & Organization

Programme(s) on which the course is given: B.Sc degree in Computer Science
Major or minor element of programmes: Major
Department offering the programme: Computer Science department
Department offering the course: Computer Science department
Academic year / Level: Third year – Second term
Date of specification approval:

A- Basic Information

Title: Computer Architecture & Organization **Code:** CSC 362
Credit Hours: -----
Lecture: 3 hrs / week
Tutorial: 0 hrs / week
Practical: 2 hrs / week
Total: 5 hrs / week

B- Professional Information

1. Overall Aims of Course.

This is an introductory course in computer organization and architecture. The objectives of this course are to introduce the student to the internal workings of the computer at both a hardware and software level. To that end, the student learns binary representations for numbers, an introduction to Boolean logic and digital circuits to understand how computer functions are carried out at the hardware level, and computer organization topics such as the role and structure of the CPU, memory, I/O system, bus and interrupt system. The student is also introduced to a basic instruction set to see how these activities are described at a programming level.

2. Intended Learning Outcomes of Course (ILOs)

a) Knowledge and Understanding:

The course aims to give the student:
a1- A good understanding of the fundamental principles for computer architectures and organizations.
a2- An understanding of the skills for designing microprocessors

b) Intellectual Skills:

At the end of the course, the student will know:
b1- How to think in the basic organization and architecture of digital computers (CPU, memory, I/O, software).

b2- How to think in Interconnection Networks and Microprocessors.

c) Professional and Practical Skills:

At the end of the course, the student will be able to:

c1- Understand the architecture of computer systems and associated software.

c2- Deal with the addressing modes, interrupt systems, input/output systems, and external memory systems.

c3- Interact with the assemblers, loaders, multiprogramming, performance evaluation, and data security.

d) General and Transferable Skills:

At the end of the course, the student will have:

d1- The ability to interact with various computer systems organization.

d2- The ability to benefit from different resources of the computer system (CPU, memory, I/O, software).

3. Content of Course

Topic	No. of Hrs	Lecture	Tutorial/ Practical
Fundamentals of Computer Design: - Introduction - The Task of a Computer Designer - Technology and Computer Usage Trends - Cost and Trends in Cost - Measuring and Reporting Performance - Quantitative Principles of Computer Design - Putting It All Together: The Concept of Memory Hierarchy - Fallacies and Pitfalls - Concluding Remarks	5	3	2
Instruction Set Principles and Examples: - Introduction - Classifying Instruction Set Architectures - Memory Addressing - Operations in the Instruction Set - Type and Size of Operands - Encoding an Instruction Set - Crosscutting Issues: The Role of Compilers - Putting It All Together: The DLX Architecture - Fallacies and Pitfalls - Concluding Remarks	5	3	2
Pipelining: - What Is Pipelining? - The Basic Pipeline for DLX - The Major Hurdle of Pipelining	5	3	2

- Pipeline Hazards - Data Hazards - Control Hazards - What Makes Pipelining Hard to Implement? - Extending the DLX Pipeline to Handle Multi-cycle Operations - Crosscutting Issues: Instruction Set Design and Pipelining - Putting It All Together: The MIPS R4000 Pipeline - Fallacies and Pitfalls - Concluding Remarks			
Advanced Pipelining and Instruction-Level Parallelism: - Instruction-Level Parallelism: Concepts and Challenges - Overcoming Data Hazards with Dynamic Scheduling - Reducing Branch Penalties with Dynamic Hardware Prediction - Taking Advantage of More ILP with Multiple Issue - Compiler Support for Exploiting ILP - Hardware Support for Extracting More Parallelism - Studies of ILP - Putting It All Together: The PowerPC 620 - Fallacies and Pitfalls - Concluding Remarks	5	3	2
Memory-Hierarchy Design: - Introduction - The ABCs of Caches - Reducing Cache Misses - Reducing Cache Miss Penalty - Reducing Hit Time - Main Memory - Virtual Memory - Protection and Examples of Virtual Memory - Crosscutting Issues in the Design of Memory Hierarchies - Putting It All Together: * The Alpha AXP 21064 Memory Hierarchy - Fallacies and Pitfalls - Concluding Remarks	5	3	2
Storage Systems:	5	3	2

- Introduction - Types of Storage Devices - Buses—Connecting I/O Devices to CPU/Memory - I/O Performance Measures - Reliability, Availability, and RAID - Crosscutting Issues: Interfacing to an Operating System - Designing an I/O System - Putting It All Together: UNIX File System Performance - Fallacies and Pitfalls - Concluding Remarks			
Interconnection Networks: - Introduction - A Simple Network - Connecting the Interconnection Network to the Computer. - Interconnection Network Media. - Connecting More Than Two Computers - Practical Issues for Commercial Interconnection Networks. - Examples of Interconnection Networks - Crosscutting Issues for Interconnection Networks. - Internetworking. - Putting It All Together: An ATM Network of Workstations - Fallacies and Pitfalls - Concluding Remarks	5	3	2
Multiprocessors: - Introduction. - Characteristics of Application Domains - Centralized Shared-Memory Architectures - Distributed Shared-Memory Architectures - Synchronization - Models of Memory Consistency - Crosscutting Issues - Putting It All Together: * The SGI Challenge Multiprocessor - Fallacies and Pitfalls - Concluding Remarks	5	3	2
Computer Arithmetic: - Introduction. - Basic Techniques of Integer Arithmetic - Floating Point	5	3	2

- Floating-Point Multiplication - Floating-Point Addition - Division and Remainder - More on Floating-Point Arithmetic - Speeding Up Integer Addition - Speeding Up Integer Multiplication and Division. - Putting It All Together. - Fallacies and Pitfalls.			
Vector Processors: - Why Vector Processors? - Basic Vector Architecture - Two Real-World Issues: Vector Length and Stride. - Effectiveness of Compiler Vectorization - Enhancing Vector Performance - Putting It All Together: Performance of Vector Processors. - Fallacies and Pitfalls - Concluding Remarks.	5	3	2
Survey of RISC Architectures: - Introduction - Addressing Modes and Instruction Formats - Instructions: The DLX Subset - Instructions: Common Extensions to DLX - Instructions Unique to MIPS - Instructions Unique to SPARC - Instructions Unique to PowerPC - Instructions Unique to PA-RISC - Concluding Remarks.	5	3	2

4. Teaching and Learning Methods

- Lectures
- Practical lab work
- Independent Work

5. Student Assessment

a) Assessment Methods

- Assignments and Quizzes
- Midterm written exam
- Oral exam
- Practical exam
- Final written exam

b) Assessment schedule

Midterm examination Week 7

Practical examination	Week 13
Oral examination	Week 14
Final examination	Week 15

c) Weighting of assessments

Assignments and Quizzes	0 %
Mid-Term Examination	10%
Oral Examination	10%
Practical Examination	15%
Final-term Examination	65%
Total	**100 %**

6. List of References

a) Recommended Books:

- David A. Patterson and John L. Hennessy, "Computer Organization & Design", 3rd edition, Morgan Kaufmann, ISBN 1558604286, 2004.
- Linda Null and Julia Lobur, "The Essentials of Computer Organization and Architecture", 2nd edition, Jones and Bartlett, ISBN 0-7637-0444-X, 2006.

7. Facilities Required for Teaching and Learning

- Computer lab.
- Data show device.

Course coordinator:

Head of Department:

Date: / /