Arab Academy for Science and Technology and Maritime Transport
Computer Science Curriculum
Course Syllabus

Course Code: CE243	**Course Title:** Introduction to Computer Architecture	**Classification:**	**Coordinator's Name:** Prof. Aliaa Youssif **Lecturer:** Prof. Aliaa Youssif	**Credit:** 3
Pre-requisites: CE216-Digital Logic Design	**Co-requisites:** None	**Schedule:** Lecture 2 hrs. Tutorial/Lab 2 hrs.		
Office Hours:				
Course Description: The course Includes the organization and architecture of computer systems hardware; instruction set architectures; addressing modes; register transfer notation; processor design and computer arithmetic; memory systems; hardware implementations of virtual memory, and input/output control and devices This course exposes the student to computer design & organization. It aims for the student to understand the software/hardware interface, instructions, processor, modules & performance issues.				
Textbook: Null Linda and Julia Lobur, *The Essentials of Computer Organization and Architecture,* Jones and Bartlett Publishers.				

References:

1. David A. Patterson, John L. Hennessy, *Computer Organization and Design: The Hardware/Software Interface*, Morgan Kaufmann.
2. Morris Mano, *Computer System Architecture*, Prentice Hall.
3. John L. Hennessy, David A. Patterson, *Computer Architecture A Quantitative Approach*, Morgan Kaufmann.

Course Objective/Course Learning Out-come:	**Contribution to Program Student Out-comes:**
1. Present the various digital components used in the organization and design of digital computers.	**(SO1)** Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions. **(SO6)** Apply computer science theory and software development fundamentals to produce computing-based solutions.
2. Explain the detailed steps that a designer must go through in order to design an elementary basic computer.	
3. Introduce the organization and architecture of the main units of a digital computer.	
4. Experiment with assembly language programming for a hypothetical machine	**(SO2)** Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline.

Course Outline: **Week 1.** Introduction **Week 2.** CPU basics and Organization **Week 3.** MARIE **Week 4.** Instruction Processing **Week 5.** Extending Instruction Set **Week 6.** Instruction Formats **Week 7.** 7th Week Exam **Week 8.** Instruction Types and Addressing **Week 9.** Instruction Pipelining and real world examples of ISAs.	**Week 10.** Types of Memory **Week 11.** Memory Hierarchy and Cache Memory **Week 12.** 12th Week Exam **Week 13.** Virtual Memory **Week 14.** Performance Measurement and analysis **Week 15.** Revision **Week 16.** Final Exam
Grade Distribution: **7th Week Assessment (30%):** Exam (20%) + Section Quiz 1 (10%) **12th Week Assessment (20%):** Exam (20%) **Year Work (10%):** Year Work (10%) **Final Exam (40%)**	

Policies:

Attendance:
AASTMT Education and Study Regulations (available at aast.edu)

Academic Honesty:
AASTMT Education and Study Regulations (available at aast.edu)

Late Submission:
Late submissions are graded out of 75% (1 week late), 50% (2 weeks late), 25% (3 weeks late), 0% (more than 3 weeks late)