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

| | | | | |
|---|---|---|---|---|
| **Course Code:** CS427 | **Course Title:** Embedded Systems Programming | **Classification:** E | **Coordinator's Name:** Dr. Essam Elfakhrany | **Credit Hours:** 3 |
| **Pre-requisites:**<br>• CE243 (Introduction to Computer Architecture)<br>• CS143 (Introduction to Problem Solving and Programming) | **Co-requisites:** None | **Schedule:**<br>Lecture: 2 hours<br>Tutorial-Lab: 2 hours | | |
| **Office Hours:** | | | | |

**Course Description:**

This course introduces students to principles, methodologies, and procedures for embedded systems. The goal is to learn how to bring hardware (microprocessors, Raspberry PI, Arduino, sensors, displays, etc.) and software (programming language, operating system (windows 10, Linux,... etc)) together to specify, design, and implement system solutions to producing whole, complete products.

**Textbook:**

Daniel W. Lewis, Fundamentals of Embedded Software with the ARM Cortex-M3, Pearson (2013).

**References:**

- Jeremy Blum, *Exploring Arduino: Tools and Techniques for Engineering Wizardry*, Wiley.
- Monk Simon, Programming the Raspberry Pi: *Getting Started with Python, Second Edition, McGraw-Hill.*

| Course Objective/Course Learning Outcome: | Contribution to Program Student Outcomes: |
|---|---|
| **1** Identify and appreciate the meaning of embedded systems applications. | (SO1) Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions. |
| 2 Use Microcontroller software with high level programming to develop task oriented applications in real time | (SO6) Apply computer science theory and software development fundamentals to produce computing-based solutions. |
| 3 Understand and use real time operating system functions and tools for multi-tasking, inter-networking, memory limitations, hardware resource | |

| | |
|---|---|
| allocation, etc… | |
| **Course Outline:**<br>1. Understanding of embedded systems using modular design and abstraction<br>2. C programming: considering both function and style<br>3. How to build and test circuits with switches, LEDs, resistors, potentiometers, and liquid crystal displays<br>4. Synchronization of hardware and software input/output with switches, lights, sound, sensors. motors, and liquid crystal displays | 5. Implementation of an I/O driver and multi-threaded programming using interrupts<br>6. How to read a data sheet of IC's, Raspberry PI, Arduino, ..etc.<br>7. How to construct a smart object and create a system as part of the *Internet of Things* |
| Grade Distribution:<br>7th Week Assessment (30%)<br>12th Week Assessment (20%)<br>Year Work (10%)<br>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)*