**COURSE DESCRIPTION FORM: EE-2003Computer Organization & Assembly Language (COAL)**

**COURSE DESCRIPTION FORM**

**INSTITUTION**  FAST School of Computing, National University of Computer and Emerging Sciences, Karachi

BS-School of Computing– Spring 2022

**PROGRAM TO BE EVALUATED**

**Course Description**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Course Code** | EE2003 | | | |
| **Course Title** | Computer Organization & Assembly Language | | | |
| **Credit Hours** | 3 | | | |
| **Prerequisites by Course(s) and Topics** | PF, DLD | | | |
| **Grading Policy** | Absolute grading | | | |
| **Policy about missed assessment items in the course** | Retake of missed assessment items (other than midterm/ final exam) will not be held.  For a missed midterm/ final exam, an exam re-take/ pre-take application along with necessary evidence are required to be submitted to the department secretary. The examination assessment and retake committee will decide the exam re-take/ pre-take cases. | | | |
| **Course Plagiarism Policy** | Plagiarism in project or midterm/ final exam may result in F grade in the course.  Plagiarism in an assignment will result in zero marks in the **whole assignments** category. | | | |
| **Assessment Instruments with Weights** (homework, quizzes, midterms, final, programming assignments, lab work, etc.) | 50% Theory 50% Practical Assessment Items   |  |  |  | | --- | --- | --- | | **Assessment Item** | **Number** | **Weight (%)** | | Assignment | 3 | 10%= 2\*3.5+3 | | Quiz | 3 | 10%= 2\*3.5+3 | | Midterm Exam | 2 | 30%= 2\*15 | | Final Exam | 1 | 50% | | | | |
| **Course Instructors** | Atiya and Shoaib Rauf | | | |
| **Lab Instructors (if any)** |  | | | |
| **Course Coordinator** | Ms. Atiya | | | |
| **URL (if any)** |  | | | |
| **Current Catalog Description** | * Programming Methodology of low-level languages * How to access computer hardware directly * Overview of a user-visible architecture (of Intel 80x86 processors) * Intel 80x86 instruction set, assembler directives, macro, etc. * How programs interact with the operating system for various services including memory management and input/output services * How is it possible to interface high-level language and low-level language modules | | | |
| **Textbook** (or **Laboratory Manual** for Laboratory Courses) | Assembly Language for Intel Based Computers K.Irvine 7th Edition  MIPS Assembly Language Programming by Ed Jorgensen, Version 1.1.35  April 2018 | | | |
| **Reference Material** | Computer organization and design: the hardware/software interface by David A. Patterson and John L. Hennessy  Computer Organization & Embedded Systems Hamacher et al. 6th Ed. | | | |
| **Course Learning Outcomes** | |  | | --- | | **A. Course Learning Outcomes (CLOs)** | | On successful completion of this course students will have to know how of:   |  |  | | --- | --- | | **B. Program Learning Outcomes** | | | For each attribute below, indicate whether this attribute is covered in this course or not. Leave the cell blank if the enablement is little or non-existent. | | | |  |  | | --- | --- | | 1. Computing Knowledge | Apply knowledge of mathematics, natural sciences, computing fundamentals, and a computing specialization to the solution of complex computing problems. | |  | | |  |  | | --- | --- | | 2. Problem Analysis | Identify, formulate, research literature, and analyze complex computing problems, reaching substantiated conclusions using first principles of mathematics, natural sciences, and computing sciences. | | ✓ | | |  |  | | --- | --- | | 3. Design/ Develop Solutions | Design solutions for complex computing problems and design systems, components, and processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations. | | ✓ | | |  |  | | --- | --- | | 4. Investigation & Experimentation | Conduct investigation of complex computing problems using research-based knowledge and research-based methods. | |  | | |  |  | | --- | --- | | 5. Modern Tool Usage | Create, select, and apply appropriate techniques, resources and modern computing tools, including prediction and modeling for complex computing problems. | |  | | |  |  | | --- | --- | | 6. Society Responsibility | Apply reasoning informed by contextual knowledge to assess societal, health, safety, legal, and cultural issues relevant to context of complex computing problems. | |  | | |  |  | | --- | --- | | 7. Environment and Sustainability | Understand and evaluate sustainability and impact of professional computing work in the solution of complex computing problems. | |  | | |  |  | | --- | --- | | 8. Ethics | Apply ethical principles and commit to professional ethics and responsibilities and norms of computing practice. | |  | | |  |  | | --- | --- | | 9. Individual and Teamwork | Function effectively as an individual, and as a member or leader in diverse teams and in multi-disciplinary settings. | |  | | |  |  | | --- | --- | | 10. Communication | Communicate effectively on complex computing activities with the computing community and with society at large. | |  | | |  |  | | --- | --- | | 11. Project Management and Finance | Demonstrate knowledge and understanding of management principles and economic decision making and apply these to one’s own work as a member or a team. | |  | | |  |  | | --- | --- | | 12. Lifelong Learning | Recognize the need for and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological changes. | |  | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **C. Mapping of CLOs on PLOs**  (CLO: Course Learning Outcome, PLOs: Program Learning Outcomes) | | | | | | | | | | | | | | | |  | | **PLOs** | | | | | | | | | | | | | | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | | **CLOs** | 1 |  |  |  |  | ✓ |  |  |  |  |  |  |  | | 2 |  | ✓ | ✓ |  |  |  |  |  |  |  |  |  | | 3 |  |  |  |  | ✓ |  |  |  |  |  |  |  | | | | | |
| **Topics covered in the course with number of lectures on each topic** (Assume 15 weeks of instruction and 1 hour lecture duration) | |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | |  | **Topics to be covered** | | | | | | **List of Topics** | | **Week** | **No. of Weeks** | **Contact Hours** | **CLO(s)** | | **Introduction**:  Introduction to Computer Architecture & Organization & Assembly Language***(1 Lecture)***  ***-----------------------------------------------------***  Applications of Assembly Language, Assemble-Link-Execute Cycle ***(1 Lecture)***  ***-----------------------------------------------------***  Assembly Relativity, Portability, Virtual Machine Concept and Machine Levels  ***(1 Lecture)*** | | **1** | **1** | **3** | **1** | | Microcomputer Concepts, Components of Microcomputer (***1 Lecture)***  ***-----------------------------------------------------***  Intel 80x86 Processor Architecture, Mode of Operations ***(1 Lecture)***  ***-----------------------------------------------------***  Basic Execution Environment  ***(1 Lecture)*** | | **2** | **1** | **3** | **1** | | Assembly Language Fundamentals: Integer, Character & String Literals, Identifier, Directive Vs Instruction (1 Lecture) ***--------------------------------------------------***  *Instruction, Defining Data*  ***(1 Lecture)***  ***--------------------------------------------------***  Symbolic Constants (1 Lecture) **Assignment no 1 Release**  **(Start of Week 3)** | | **3** | **1** | **3** | **2** | | Data Transfer (1 Lecture) ***--------------------------------------------------*** Addressing(1 Lecture) ***--------------------------------------------------*** Arithmetic Operations(1 Lecture) **Assignment no 1 Submission**  **(End of Week 4)** | | **4** | **1** | **3** | **2** | | Operators and Directive (1 Lecture) ***--------------------------------------------------*** Instruction to control transfer Instructions(1 Lecture) ***--------------------------------------------------*** Arrays and Loops(1 Lecture) | | **5** | **1** | **3** | **2** | | ***WEEK 6*** | ***MID -1 Exam*** | | | | | | Procedures and Stack Operations (1 Lecture) ***--------------------------------------------------*** Runtime Stack(1 Lecture) ***--------------------------------------------------*** PUSH and POP Instructions(1 Lecture) **Assignment no 2 Release**  **(Start of Week 7)** | | **7** | **1** | **3** | **4** | | **Conditional Processing:** Boolean and comparison instruction, conditional jumps ***(3 Lectures)*** | | **8** | **1** | **3** | **2** | | conditional loop structures, high-level language constructs***(3 Lectures)***  **Assignment no 2 Submission**  **(End of Week 9)** | | **9** | **1** | **3** | **2** | | Shift & Rotate *Instructions****(1 Lectures)***  ***--------------------------------------------------***  Multiplication & Division instructions  ***(1 Lecture)***  ***--------------------------------------------------***  Extended Addition & Subtraction***(1 Lecture)*** | | **10** | **1** | **3** | **2** | | ***Week 11*** | ***MID -2 Exam*** | | | | | | **Advanced Procedures – Introduction and Examples:**Stack Frames  ***--------------------------------------------------*** Recursion(1 Lecture) ***--------------------------------------------------*** INVOKE, ADDR, PROC, PROTO Directives (1 Lecture) **Assignment no 3 Release**  **(Start of Week 12)** | | **12** | **1** | **3** | **1,2,4** | | **String and Arrays**  String primitive Instructions  ***(3 Lectures)***  ***--------------------------------------------------*** Two dimensional array(1 Lecture) **Assignment no 3 Submission**  **(End of Week 13)** | | **13** | **1** | **3** | **2** | | **Machine Language Translation**  Instruction Formats, encoding an Instruction Set and Modes of Addressing, Translation and Working of an Assembler, Map File and Memory Map  ***(3 Lectures)*** | | **14** | **1** | **3** | **3** | | CISC vs RISC, Introduction to MIPS Assembly***(3 Lectures)*** | | **15** | **1** | **3** |  | | ***Week 16*** | ***Final Exam*** | | | | | | **Review** | |  | **1** | **3** |  | | *Total* | |  | **16** | **48** |  | | | | |
| **Laboratory Projects/Experiments Done in the Course** | **Mentioned in Lab Course Description** | | | |
| **Programming Assignments Done in the Course** | 3 Assignments are given which are attached in the assignments section | | | |
| **Class Time Spent** (in percentage) | **Theory (%)** | **Problem Analysis (%)** | **Solution Design (%)** | **Social and Ethical Issues (%)** |
| 50 | 25 | 20 | 5 |
| **Oral and Written Communications** | Every student is required to submit at least 1 written report of typically 10 pages in IEEE research report format. Students will also be called for viva/presentation of the project and any assignment where necessary in Lab Section | | | |

**Instructor Name: Atiya Jokhio**

**Instructor Signature: *Atiya***

**Date: 1st-Feb-2022**