Resource Person:

Semester:

Course Title: Computer Organization and Architecture

Course Code:

Course Type: Core

Pre-Requisite: Digital Logic and Design

Counseling Hours:

Program: MCS

Program Head: Imran Saleem

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Name** | **Signature** | **Date** |
| **Checked By**(Program Head) |  |  |  |
| **Approved By** (Head SPA) |  |  |  |

**Course Description & Format**

This course is divided in three main Modules. The first module addresses the theoretical issues. Practical skills will be inculcated through in-class activities in the second session in which student should be able to write an Assembly code and know-how of the Computer Components, hierarchy and devices work together.

This Course is aimed to give an overview of Organization and Architecture of Computer to a student that how it is built. It Provides the brief discussion of CPU, MIPS instruction, performance, Memory, I/O , Data-paths, Hazards, Control Units, Logic Gates, Number formats, Registers and Flags, Pipelining, organization and peripherals. The later session will identify the various hierarchical views of a computer as gates, microprogram, machine language, assembly language, Write and debug code in assembly language.

At the end student should know the different components and devices of a computer system and learn how these devices and components work together in an efficient way. Upon Successful completion of Course student will be able to describe the Logic Gates and theory behind them. Understand CPU Design, performance and Evaluation, its instruction like fetch -decode etc. Student should be able to distinguish between Arithmetic and logical Operators. Student should understand the Memory System like Cache, Virtual Memory and Introduction to registers and flags. Later on student should have understanding of Hazards, Pipelined Processor and Data path. At the end Student should be able to translate high level language code into machine language and assembly.

**Course Instructional Objectives**

* The main instructional goal is to inculcate theoretical and conceptual framework of Computer Organization and Assembly.
* This will be achieved through lectures, in-class activities and corporate sessions.
* Introduce the organization of computer systems and usage of Assembly Language for optimization and Control.
* Emphases should be given to expose the low level logic for problem solving while Assembly Language use as a Tool.

**Course Student Objectives**

* Students should be able to describe the basic functionalities and limitations of computer systems.
* Students will learn the theoretical and conceptual framework of Computer Organization and Assembly Language.
* The main objective however is to develop capabilities and competencies in learning this course.
* After undertaking this course, the students would be able to conduct Performance and Evaluation on their own.

**Course Contents**

Following is the session-wise breakup of the course:

**Session 1: Introduction**

* One-to-one introduction
* Course Introduction, Teaching & Assessment Methodology
* Distribution of Course Outlines
* Discussion on Course Outline
* Setting up of Norms
* Dividing the class in 4 to 5 Study Groups which will remain till the completion of the course
* A Brief introduction to Computer Organization and Assembly

**Session 2: Computer Abstractions and Technology**

* Introduction
* Eight Great Ideas in Computer Architecture
* Below Your Program
* Under the Covers
* Technologies for Building Processors and Memory
* Performance

**Learning Outcomes**

The main learning objective of the reading is to provide a basic and conceptual understanding of Computer Organization and Assembly along with learning binary number systems, gate-sequencing and performance evaluations.

**Activities:**

* 1st Assignment

**Session 3: Instructions: Language of the Computer**

* Introduction
* Operations of the Computer Hardware
* Operands of the Computer Hardware
* Signed and Unsigned Numbers
* Representing Instructions in the Computer

**Learning Outcomes**

This session provides a learning of operations and operands related to computer hardware, signed and unsigned numbers and instruction representation in computer systems.

**Activities:**

* Class Activity

**Session 4: Instructions: Language of the Computer**

* Logical Operations
* Bit manipulations
* Bitwise logical operations and masking operations
* Instructions for Making Decisions

**Learning Outcomes**

Basics of assembly language and basic conversion of codes

**Activities:**

* 1st Quiz
* 2nd Assignment

**Session 5: Arithmetic for Computers**

* Introduction
* Addition and Subtraction
* Multiplication
* Division
* Floating point operations

**Learning Outcomes**

Students will be trained in performing the all Arithmetic and logical operations like Addition,CLA etc. Individual assessment will be done as a Class Activity.

**Activities:**

* Class Activity

**Session 6: Assembly Programming**

* Assembly language code
* Addressing modes of MIPS instructions
* Assembly language programs with MIPS

**Learning Outcomes**

Student should be able to write and debug code in assembly. Expected outcome of this session is that students should have eligibility to translate higher language code like C to assembly and machine Language.

**Activities:**

* Lab Work

**Session 7: Assembly Programming**

* Subroutines
* Stack and its operations
* Parameter Passing through stack and local variables
* Display memory and number printing in assembly

**Learning Outcomes**

Student should be able to perform the subroutines and the operations of stack in assembly language. Also they will learn how to display memory and number printing in assembly language.

**Activities:**

* 2nd Quiz
* Lab Work
* Revision Session

**Session 8: MIDTERM**

**Session 9: The Processor**

* Logic Design Conventions
* Processor Design
* Instruction set design and addressing
* Building a Data path

**Learning Outcomes**

Student should be able to understand step by step procedure of building a processor. Understand data paths of different instructions set design and the addressing.

**Activities:**

* Midterm solution discussed and reviewed
* 3rd Assignment

**Session 10: Processor Performance**

* Performance benchmarks
* Power and energy
* Latency and throughput
* Ahmdal’s Law
* Comparison of Processors

**Learning Outcomes**

In this session students will learn about different benchmarks used to measure performance of a processor along with the different factors affecting it.

**Activities:**

* 3rd Quiz

**Session 11: Pipelining**

* An Overview of Pipelining
* Pipelining of processor issues and hurdles
* Pipelined Data path and Control
* Hazards
* Static branch prediction

**Learning Outcomes**

Student should be able to pipelining processes and data path related to pipelining, the limitations of pipelining and the solutions to these limitations.

**Activities:**

* 4th Assignment
* Class Activity

**Session 12: Multi-cycle Pipelines, Exceptions**

* Multi-cycle FP MIPS Pipeline
* Exceptions handling
* Types of Exceptions
* Parallelism
* Multiprocessor systems

**Learning Outcomes**

In this session the students will learn advanced pipelining techniques in the form of multicycle pipelines and issues related to them. Additionally they will be introduced to the concept of exceptions, the reasons they occur and how to handle them.

**Activities:**

* 4th Quiz

**Session 13: Dynamic Branch Prediction**

* Basics
* 1-bit and 2-bit predictors
* m-n Bit predictors
* Limitations and enhancements

**Learning Outcomes**

The students will learn about dynamic branch prediction techniques to resolve branches during execution, how these techniques work and what are the limitations that may restrict their usage?

**Activities:**

* Class Activity
* 5th Assignment

**Session 14: Caches and Memory Organization**

* Memory organization in a computer
* Caches
* Cache structure
* Addressing and data retrieval strategies

**Learning Outcomes**

The discussion during this session will focus on the memory organization in a computer system and how caches affect the flow of data between the processor and main memory? Other topics will include addressing and retrieval of data from caches.

**Activities:**

* 5th Quiz
* Presentations

**Session 15: Presentations and Revision Session**

**Session 16: FINALTERM**

**Recommended Book (s) & Text:**

1. Computer Organization and Design, DA Patterson and JL Hennessy, 5th Edition
2. Computer Architecture - A Quantitative Approach (4th Ed) by Hennessy, J. L, and Patterson, D. A.,*– Publisher Morgan Kaufmann*
3. Computer Organization & Architecture (8th Ed) by *William Stallings*

**ASSESSMENT METHODOLOGY**

|  |  |
| --- | --- |
| Class Participation and Attendance | 10% |
| Quizzes | 10% |
| Assignments  | 10% |
| Presentation | 10% |
| Mid Term | 25% |
| Final Term Exam | 35% |
| Total | 100% |

**CALENDAR OF ACTIVITIES**

|  |  |  |  |
| --- | --- | --- | --- |
| **Session** | **Chapter** | **Readings** | **Activities** |
| 1 | **Introduction*** One-to-one introduction
* Course Introduction, Teaching & Assessment Methodology
* Distribution of Course Outlines
* Discussion on Course Outline
* Setting up of Norms
* Dividing the class in 4 to 5 Study Groups which will remain till the completion of the course
* A Brief introduction to Computer Organization and Assembly
 |  |  |
| 2 | **Computer Abstractions and Technology*** Introduction
* Eight Great Ideas in Computer Architecture
* Below Your Program
* Under the Covers
* Technologies for Building Processors and Memory
* Performance
 |  | * 1st Assignment
 |
| 3 | **Instructions: Language of the Computer*** Introduction
* Operations of the Computer Hardware
* Operands of the Computer Hardware
* Signed and Unsigned Numbers
* Representing Instructions in the Computer
 |  | * Class Activity
 |
| 4 | **Instructions: Language of the Computer*** Logical Operations
* Bit manipulations
* Bitwise logical operations and masking operations
* Instructions for Making Decisions
 |  | * 1st Quiz
* 2nd Assignment
 |
| 5 | **Arithmetic for Computers*** Introduction
* Addition and Subtraction
* Multiplication
* Division
* Floating point operations
 |  | * Class Activity
 |
| 6 | **Assembly Programming*** Assembly language code
* Addressing modes of MIPS instructions
* Assembly language programs with MIPS
 |  | * Lab Work
 |
| 7 | **Assembly Programming*** Subroutines
* Stack and its operations
* Parameter Passing through stack and local variables
* Display memory and number printing in assembly
 |  | * 2nd Quiz
* Lab Work
* Revision Session
 |
| 8 | **MIDTERM** |  |  |
| 9 | **The Processor*** Logic Design Conventions
* Processor Design
* Instruction set design and addressing
* Building a Data path
 |  | * Midterm solution discussed and reviewed
* 3rd Assignment
 |
| 10 | **Processor Performance*** Performance benchmarks
* Power and energy
* Latency and throughput
* Ahmdal’s Law
* Comparison of Processors
 |  | * 3rd Quiz
 |
| 11 | **Pipelining*** An Overview of Pipelining
* Pipelining of processor issues and hurdles
* Pipelined Data path and Control
* Hazards
* Static branch prediction
 |  | * 4th Assignment
* Class Activity
 |
| 12 | **Multi-cycle Pipelines, Exceptions** * Multi-cycle FP MIPS Pipeline
* Exceptions handling
* Types of Exceptions
* Parallelism
* Multiprocessor systems
 |  | * 4th Quiz
 |
| 13 | **Dynamic Branch Prediction*** Basics
* 1-bit and 2-bit predictors
* m-n Bit predictors
* Limitations and enhancements
 |  | * Class Activity
* 5th Assignment
 |
| 14 | **Caches and Memory Organization*** Memory organization in a computer
* Caches
* Cache structure
* Addressing and data retrieval strategies
 |  | * 5th Quiz
* Presentations
 |
| 15 | **Presentations and Revision Session**  |  |  |
| 16 | **FINALTERM** |  |  |