****

**BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI**

**WORK INTEGRATED LEARNING PROGRAMMES**

**COURSE HANDOUT**

**Part A: Content Design**

|  |  |
| --- | --- |
| **Course Title** | Computer Organization and Software Systems |
| **Course No(s)** | SS ZG516 |
| **Credit Units** | 5 (1 + 2 + 2)  Unit split between Class Hours + Lab/Design/Fieldwork + Student preparation respectively; each unit translates to 32 hours |
| **Course Author** | Lucy J Gudino / Chandra Shekar |
| **Version No** | 3.0 |
| **Date** | 17/08/2022 |

**Course Objectives**

|  |  |
| --- | --- |
| **No** | **Course Objective** |
| **CO1** | Introduce students to systems aspects ( i.e. Computer Organization and Operating Systems) involved in software development |
| **CO2** | Equip the student to understand the computer architectural and operating systems related issues that affect the performance and nature of a software |
| **CO3** | To prepare students to be in a position to evaluate/correlate high level software performance based on its system level features (i.e. architectural and operating systems) |

**Text Book(s)**

|  |  |
| --- | --- |
| T1 | Stallings William, ***Computer Organization & Architecture***, Pearson Education, 10th Ed. 2013 |
| T2 | A Silberschatz, Abraham and others, ***Operating Systems Concepts***, Wiley Student Edition, 9th Ed. |

**Reference Book(s) & other resources**

|  |  |
| --- | --- |
| R1 | Patterson, David A & J L Hennenssy, ***Computer Organization and Design – The Hardware/Software Interface***, Elsevier, Revised 4th Ed. |
| R2 | Randal E. Bryant, David R. O’Hallaron, ***Computer Systems – A Programmer’s Perspective,*** Pearson, 2nd Ed, 2016. |
| R3 | Kai Hwang and Briggs, ***Computer Architecture and Parallel*** *Processing,* Tata McGrawHill Edition |
| R4 | Stallings, ***Operating Systems: Internals and Design Principles***, International Edition, Pearson Education, 2013 (Pearson Online) |

**Modular Content Structure**

1. **Introduction to Computer Systems** 
   1. Hardware Organization of a uniprocessor computer
      1. Basic uniprocessor Architecture
      2. Instruction Cycle State Diagram
   2. Operating System role in Managing Hardware
      1. Running a Hello Program
      2. Processes, Threads, Virtual Memory, Files
   3. Performance Assessment
      1. MIPS Rate
      2. Amdahl’s Law
2. **Computer Organization and Design : Memory Organization** 
   1. Internal Memory
      1. Semiconductor Main Memory (SRAM and DRAM)
      2. DDR – DRAM
      3. Error Correction – Hamming Code
   2. External Memory
      1. Magnetic Disk ( Not in very detail)
      2. RAID
      3. SSD - Solid State Memory Technologies (Types)
3. **Computer Organization and Design :** **Cache Memory Organization**
   1. Locality
      1. Locality of Reference to Program Data
      2. Locality of instruction fetches
   2. Memory Hierarchy
   3. Cache Memories
      1. Generic Cache Memory Organization
      2. Direct-Mapped Caches
      3. Fully Associative Caches
      4. Set Associative Caches
      5. Issues with Writes
      6. Performance Impact of Cache Parameters
      7. Writing Cache friendly Codes
      8. Replacement Algorithms
   4. Processing : In-memory vs. (from) secondary storage vs over the network
4. **Computer Organization and Design : Central Processing Unit (4 hrs)**
   1. Computer Architecture and Organization
      1. Von-Neumann Architecture vs Harvard Architecture
      2. RISC Vs CISC
         1. Machine Instruction Characteristics
         2. Types of Operands, Operations and Addressing Modes
         3. Instruction Formats
         4. Hardwired vs microprogrammed control unit
      3. Case study : MIPS Single cycle implementation
   2. Pipeline
      1. Overview of pipeline
      2. Hazards: Resource, Data and Control Hazards
      3. Hazards – Mitigation
      4. Pipeline job scheduling
5. **Process Management**
   1. Concept of Process
   2. Process State Diagram
   3. Operations on Processes : Process creation and termination examples
   4. Process vs. Threads
   5. Process Scheduling criteria
   6. Process Scheduling Algorithms -FCFS, SJF, Priority, RR, Multilevel Queue, Multilevel Feedback Queue
6. **Process Coordination**
   1. The Critical section problem and Peterson’s Solution
   2. Semaphores
   3. Deadlock:
      1. System Model
      2. Deadlock Characterization

6.4. Methods of Handling Deadlocks

6.4.1 Deadlock Prevention

6.4.2 Deadlock Avoidance: Banker’s Algorithm

6.4.3 Deadlock Detection

6.4.4 Recovery from Deadlock

1. **Memory Management**
   1. Memory-Management Strategies
   2. Paging
   3. Segmentation
   4. Virtual-Memory
   5. Demand Paging
   6. Page Replacement Algorithms: FIFO, Optimal, LRU, and LFU
2. **Optimizing Program Performance**
   1. Capabilities and Limitations of Optimizing Compilers
   2. Expressing Program Performance
   3. Eliminating Loop Inefficiencies
   4. Reducing Procedure Calls
   5. Eliminating Unneeded Memory References
   6. Understanding Modern Processors
   7. Loop Unrolling
   8. Enhancing Parallelism
3. Parallel and distributed systems
   1. Motivation for parallel Processing
   2. Flynn’s classification
   3. Parallel Processing Models
   4. Shared memory vs distributed Memory
   5. Memory Hierarchy in parallel Systems
   6. Impact of Memory Hierarchy on performance
      1. Shared memory and memory contention
      2. Communication Cost
      3. Locality
   7. Memory Hierarchy in distributed systems

**Learning Outcomes:**

|  |  |
| --- | --- |
| **No** | **Learning Outcomes** |
| LO1 | Students will **apply** the knowledge of performance metrics to find the performance of systems. |
| LO2 | Students will **Examine** different computer architectures and hardware |
| LO3 | Students will **Analyse and Compare** of process management concepts including scheduling, synchronization ,deadlocks |
| LO4 | Students will **Examine** multithreading and system resources sharing among the users |
| LO5 | Students will **Analyse and Compare** various memory management techniques |
| LO6 | Students will apply different code optimization techniques |
| LO7 | Students will **Investigate** high performance architecture design |

**Part B: Contact Session Plan**

|  |  |
| --- | --- |
| **Academic Term** | SEM I, 2022-23 |
| **Course Title** | Computer Organization and Software Systems |
| **Course No** | SS \*ZG516 |
| **Lead Instructor** | Prof. Chandra Shekar RK |
| **Instructors** | Prof. Lucy J Gudino, Prof. Pradeep HK, Prof. Sarma, Prof. Chandra Shekar RK |

**Course Contents**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sl. No.** | **Contact Hour**  **#** | **List of Topic Title**  **(from content structure in Part A)** | **Topic #**  **(from content structure in Part A)** | **Text/Ref Book/external resource** |
| 1 | 1-2 | **Introduction to Computer Systems**   * Hardware Organization of a uniprocessor computer (T1: 3.1, 3.2)   + Basic uniprocessor Architecture   + Instruction Cycle State Diagram * Operating System role in Managing Hardware (T1: 8.1)   + Running a Hello Program (R2: 1.2)   + Processes, Threads, Virtual Memory, Files (Class Notes) * ***\*Performance Assessment (R1: 1.4, T1: 2.3)***    + ***MIPS Rate***   + ***Amdahl’s Law***   \* to be covered in tutorial | 1.1-1.3 | T1, R1, R2 |
| 2 | 3-4 | **Computer Organization and Design : Memory Organization**   * Internal Memory (T1: 5.1 – 5.3)   + Semiconductor Main Memory (SRAM and DRAM)   + DDR – DRAM   + Error Correction – Hamming Code * External Memory (T1: 6.1 )   + Magnetic Disk ( Not in very detail) | 1.5, 2.1 | Class Notes, T1 , R2 |
| 3 | 5-6 | **Computer Organization and Design : Memory Organization (Contd..)**   * External Memory (T1: 6.2, 6.3 )   + RAID   + SSD - Solid State Memory Technologies (Types)   **Computer Organization and Design :** **Cache Memory Organization**   * Locality (Class Notes)   + Locality of Reference to Program Data   + Locality of instruction fetches * Memory Hierarchy (T1: 4.1) * Cache Memories (T1: 4.2, 4.3)   + Generic Cache Memory Organization   + Direct Mapped Cache (T1: 4.3) | 2.2, 3.1, 3.2, 3.3 (3.3.1) | T1 |
| 4 | 7-8 | **Computer Organization and Design :** **Cache Memory Organization**   * Cache Memories (Contd..)   + Fully Associative Cache (T1: 4.3)   + Set Associative Caches (T1: 4.3)   + Issues with Writes (T1: 4.3)   + Performance Impact of Cache Parameters (T1: 4.3) | 3.3.2 – 3.3.6 | T1, R2 |
| 5 | 9-10 | **Computer Organization and Design :** **Cache Memory Organization (Contd)**   * Cache Memories (Contd..)   + Writing Cache friendly Codes (R2: 6.5)   + Replacement Algorithms (Class Notes)   + Processing : In-memory vs. (from) secondary storage vs over the network (Class Notes)   **Computer Organization and Design : Central Processing Unit**   * Computer Architecture and Organization   + Von-Neumann Architecture vs Harvard Architecture (Class Notes)   + CISC (T1: 12.1 – 12.5, 13.1 – 13.4)     - * Machine Instruction Characteristics, Types of Operands, Types of Operations, Addressing Modes, Instruction Formats   ***\* Problem Solving: Cache Memory and Replacement Algorithms (Class Notes)***  \* to be covered in tutorial | 3.3.7-3.3.8, 3.4  4.1 (4.1.1 – 4.1.2) | T1 |
| 6 | 11-12 | **Computer Organization and Design : Central Processing Unit (Contd…)**   * CISC (R1: Selected topics from Chapter 2 and Chapter 4, and Class Notes)   + Machine Instruction Characteristics, Types of Operands, Types of Operations, Addressing Modes, Instruction Formats * Hardwired vs microprogrammed control unit (Selected topics from T1: Ch 20 and 21 | 4.1 (4.1.2-4.1.3) | R1, T1 |
| 7 | 13-14 | **Computer Organization and Design : Central Processing Unit (contd..)**   * Case study: MIPS Single cycle implementation (Selected topics from R1) * Pipeline (T1: 14.4)   + Overview of pipeline   + Resource Hazard   + Data Hazard : Forwarding versus Stalling | 4.1.3 | T1, R1 |
| 8 | 15-16 | **Computer Organization and Design : Central Processing Unit (contd..)**   * Pipeline (T1: 14.4)   + Control Hazard   + Pipeline job scheduling (Class Notes) | 4.2 | T1 |
| **MID SEM EXAMINATION** | | | | |
| 9 | 17-18 | **Process Management** ( T2: 3.1-3.3, 4.1, 6.1-6.3)   * Concept of Process * Process State Diagram * Operations on Processes : Process creation and termination examples * Process vs. Threads * Process Scheduling criteria * Process Scheduling Algorithms -FCFS, SJF | 5.1-5.6 | T2 |
| 10 | 19-20 | **Process Management (Contd…)** (T2: 6.3)   * Process Scheduling Algorithms - Priority, RR, Multilevel Queue, Multilevel Feedback Queue   **Process Coordination**   * The Critical section problem and Peterson’s Solution (T2: 5.1-5.3) * Semaphores (T2: 5.6)   ***\*Problems – Scheduling Algorithm, semaphores to be covered in tutorial*** | 5.6, 6.1, 6.2 | T2 |
| 11 | 21-22 | **Process Coordination** (T2: 7.1, 7.2)   * Deadlock:   + System Model   + Deadlock Characterization * Methods of Handling Deadlocks (T2: 7.3-7.7)   + Deadlock Prevention   + Deadlock Avoidance: Banker’s Algorithm   + Deadlock Detection   + Recovery from Deadlock   ***\*Problems – deadlock to be covered in tutorial*** | 6.3 | T2 |
| 12 | 23-24 | **Memory Management** (T2: 8.1, 8.3, 8.4, 8.5, 9.1,9.2)   * Memory-Management Strategies * Paging * Segmentation * Virtual-Memory * Demand Paging   ***\*Problems – Memory Management to be covered in tutorial*** | 7.1 – 7.5 | T2 |
| 13 | 25-26 | **Memory Management (Contd…)** (T2: 9.1-9.4)   * Page Replacement Algorithms: FIFO, Optimal, LRU and LFU   **Optimizing Program Performance (R2: 5.1:5.5)**   * Capabilities and Limitations of Optimizing Compilers * Expressing Program Performance | 7.6  8.1, 8.2 | T2, R2 |
| 14 | 27-28 | **Optimizing Program Performance (R2: 5.1:5.5)**   * Eliminating Loop Inefficiencies * Reducing Procedure Calls * Eliminating Unneeded Memory References * Understanding Modern Processors * Loop Unrolling * Enhancing Parallelism | 8.3-8.8 | R2 |
| 15 | 29-30 | **Parallel and distributed systems (Selected topics from R3)**   * Motivation for parallel Processing * Flynn’s classification * Parallel Processing Models * Shared memory vs distributed Memory | 9.1-9.4 | R3 |
| 16 | 31-32 | **Parallel and distributed systems (Selected topics from R3)**   * Memory Hierarchy in parallel Systems * Impact of Memory Hierarchy on performance   + Shared memory and memory contention   + Communication Cost   + Locality * Memory Hierarchy in distributed systems | 9.5-9.6 | R3 |
| Comprehensive Examination | | | | |

**Evaluation Scheme**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Evaluation Component** | **Name**  (Quiz, Lab, Project, Midterm exam, End semester exam, etc) | **Type**  (Open book, Closed book, Online, etc.) | **Weight** | **Duration** | **Day, Date, Session, Time** |
| **EC – 1** | Quizzes / Assignment | Online | 5+25% | NA | To be announced |
| **EC – 2** | Mid-term Exam | Closed book | 30% | - | To be announced |
| **EC – 3** | End Semester Exam | Open book | 40% | - | To be announced |

***Note*** *- Evaluation components can be tailored depending on the proposed model.*

**Important Information**

Syllabus for Mid-Semester Test (Closed Book): Topics in Weeks 1-8 (1-18 Hours)

Syllabus for Comprehensive Exam (Open Book): All topics given in plan of study

Evaluation Guidelines:

1. EC-1 consists of either two Assignments or three Quizzes. Announcements regarding the same will be made in a timely manner.
2. For Closed Book tests: No books or reference material of any kind will be permitted. Laptops/Mobiles of any kind are not allowed. Exchange of any material is not allowed.
3. For Open Book exams: Use of prescribed and reference text books, in original (not photocopies) is permitted. Class notes/slides as reference material in filed or bound form is permitted. However, loose sheets of paper will not be allowed. Use of calculators is permitted in all exams. Laptops/Mobiles of any kind are not allowed. Exchange of any material is not allowed.
4. If a student is unable to appear for the Regular Test/Exam due to genuine exigencies, the student should follow the procedure to apply for the Make-Up Test/Exam. The genuineness of the reason for absence in the Regular Exam shall be assessed prior to giving permission to appear for the Make-up Exam. Make-Up Test/Exam will be conducted only at selected exam centres on the dates to be announced later.

It shall be the responsibility of the individual student to be regular in maintaining the self-study schedule as given in the course handout, attend the lectures, and take all the prescribed evaluation components such as Assignment/Quiz, Mid-Semester Test and Comprehensive Exam according to the evaluation scheme provided in the handout.