|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **NAME OF**  **COURSE** | **DURATION** | **BATCH** | **BOS DATED** | **STATUS** |
| MCA | 2 YEARS | WEF 2023 | **10 JUNE 2023** | ORIGINAL |

MASTER OF COMPUTER APPLICATION (MCA)

**Introduction to the course**

## India has proved its natural expertise in software development and related expertise after emergency as largest BPO organizations to cater top IT industries worldwide. Now this natural quality of Indians to outperform in logic development has made them as a natural choice for HR recruitment in any big IT organization. This process is not only creating jobs for Indians worldwide but also enhancing the foreign reserve level of our nation with this situation in mind country needs excellent education facilities to generate a HR of high quality to cater IT needs worldwide.

**Program Objective**

## To prepare graduates who will be successful in industry, government, academia, research, entrepreneurial pursuit and consulting firm

1. Student will develop software solutions to problems across a broad range of application domain.

## Graduates will be able to communicate technical information effectively, both orally and in writing

1. Graduates will be able to work collaboratively as a member or leader in multidisciplinary teams.

**Program Outcome**

|  |  |
| --- | --- |
| **No.** | **Program Outcome** |
| PO1 | Computational Knowledge: Apply knowledge of computing fundamentals, computing specialization, mathematics, and domain knowledge appropriate for the computing specialization to the abstraction and conceptualization of computing models from defined  problems and requirements. |
| PO2 | Problem Analysis: Identify, formulate, research literature, and solve complex computing  problems reaching substantiated conclusions using fundamental principles of mathematics, computing sciences, and relevant domain disciplines. |



|  |  |
| --- | --- |
| PO3 | Design /Development of Solutions: Design and evaluate solutions for complex computing problems, and design and evaluate systems, components, or processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and  environmental considerations. |
| PO4 | Conduct investigations of complex Computing problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and  synthesis of the information to provide valid conclusions. |
| PO5 | Modern Tool Usage: Create, select, adapt and apply appropriate techniques, resources, and  modern computing tools to complex computing activities, with an understanding of the limitations. |
| PO6 | Professional Ethics: Understand and commit to professional ethics and cyber regulations, responsibilities, and norms of professional computing practices. |
| PO7 | Life-long Learning: Recognize the need, and have the ability, to engage in independent learning  for continual development as a computing professional. |
| PO8 | Project management and finance: Demonstrate knowledge and understanding of the computing and management principles and apply these to one’s own work, as a member and  leader in a team, to manage projects and in multidisciplinary environments. |
| PO9 | Communication Efficacy: Communicate effectively with the computing community, and with society at large, about complex computing activities by being able to comprehend and write effective reports, design documentation, make effective presentations, and give and understand  clear instructions. |
| PO10 | Societal and Environmental Concern: Understand and assess societal, environmental, health, safety, legal, and cultural issues within local and global contexts, and the consequential  responsibilities relevant to professional computing practices. |
| PO11 | Individual and Team Work: Function effectively as an individual and as a member or leader in  diverse teams and in multidisciplinary environments. |
| PO12 | Innovation and Entrepreneurship: Identify a timely opportunity and using innovation to pursue that opportunity to create value and wealth for the betterment of the individual and society at  large. |

1. **Scope and Content:**

## The Regulation documented here are applicable for the Master of Computer Applications programme offered by the University.

* 1. The applicability of the Regulation must be understood in the context of the given Scheme of study and Syllabus of the programme.



## The authorities of university may modify, add, delete expand or substantiate any part of the Regulations and syllabi, at any time.

1. **Course Content:**

The programme shall be for the duration of six semesters, spread out in three years. Each semester of the programme shall consist of either all or some of the following components:

# Core Subjects

## **AECC** (Ability Enhancement Compulsory Course)

* 1. **SEC** (Skill Enhancement Course)

## **DSE** (Discipline Specific Electives)/Choice Based

* 1. **GE** (Generic Electives)

## Lab Course

* 1. Project Work
  2. **Core Subjects**

Core subjects comprise of subjects that form an integral part of the programme. These subjects provide a strong ground in basic disciplines of study.

* 1. **AECC (Ability Enhancement Compulsory Course)**

The students who have not done English up to class XII are to opt for Hindi Communication. They can opt Environment studies and other languages also.

* 1. **SEC (Skill Enhancement Course)**

This will facilitate student mobility across institutions within and across countries and also enable potential employers to assess the performance of students.

* 1. **DSE (Discipline Specific Electives)/Choice Based**

Elective courses may be offered by the main discipline/subject of study is referred to as Discipline Specific Elective. The University/Institute may also offer discipline related Elective courses of interdisciplinary nature (to be offered by main discipline/subject of study)

* 1. **GE (Generic Electives)**



An elective course chosen generally from an unrelated discipline/subject, with an intention to seek exposure is called a Generic Elective. P.S.: A core course offered in a discipline/subject may be treated as an elective by other discipline/subject and vice versa and such electives may also be referred to as Generic Elective.

* 1. **Lab Courses**

These subjects are totally practical-based subjects. The learning of these subjects will be performed in laboratories/practical sites with equipment’s/resources. These subjects shall support the practical implementation of the core/core-bracket subjects. The processes of evaluation of their subjects will depend on the nature of that individual subject.

* 1. **Project Work**

The project work shall be done for duration as specified by the coordinator, in the area, related to the main subject of study or the specialization. The project work shall give the student an insight to the situations existing in the field/marked/industries, etc.

1. **Eligibility for admission and mode of selection**
   1. The minimum qualification required to be eligible for admission is passed BCA / bachelor’s degree in computer science engineering or Equivalent Degree. **OR** Passed B.Sc./B.Com./B.A. with **Mathematics** at 10+2 level or at Graduation Level (with additional Bridge courses as per the norms of the concerned University). Obtained at least 50% marks (45% marks in case of candidates belonging to reserved category) in the qualifying examination.
   2. The method of selection for the course shall normally be by means of an entrance test and a personal interview.
2. **Attendance Requirement**

A student is eligible to appear for the term-end examinations, only if she/he has put in a minimum of 75% attendance in each subject individually.

1. **Assessment and Examination**

# Credits

Credit Points will be awarded for all the subjects. One credit is equivalent to ten classroom contact hours.Each core subjects will carry either 4 or 2 credits, each bracket subject will carry 3 credits and practical courses will carry either 4 or 2 credits depending on the number of hours of teaching and training.

# Pattern of Assessment

* Assessment of student’s performance will be based on two components i.e. Internal Assessment and Term-end Examination conducted at the end of each semester.



* A four-credit subject will comprise of an Internal Assessment component of 30 marks and a Term-end Examination components of 70 marks.
* A two-credit subject will comprise of an Internal Assessment component of 15 marks and a Term-end Examination components of 35 marks.
* Sessional tests, assignments, mid-term examinations etc, will be conducted in each subject during the course of each semester, for the purpose of internal assessment.
* The Term-end examinations will be conducted as per the University regulations.

# Assessment for Core Bracket subjects

Depending on the participation and performance of students, the faculty of the Core Bracket subject will grade the student in terms of a four-point scale as given below:

|  |  |  |
| --- | --- | --- |
| **Marks Secured** | **Grade**  **Point** | **Letter Grade** |
| 80 and above | 10 | Outstanding(O) |
| 70 and above but below 80 | 9 | Excellent (A+) |
| 65 and above but below 70 | 8 | Very Good (A) |
| 60 and above but below 65 | 7 | Good (B+) |
| 55 and above but below 60 | 6 | Above Average (B) |
| 50 and above but below 55 | 5 | Average (C) |
| 45 and above but below 50 | 4 | Pass(P) |
| Below 45 | 0 | Fail (F) |
|  | 0 | Absent (AB) |

This assessment is purely based on internal assessment of the subject faculty/coordinator.

# Assessment of Project work

The project work will carry a total of 600 marks. Of this, 70% marks are for the external examination and 30% marks will be awarded for internal evaluation.

**Evaluation of Industrial Project and Seminar:**

During the 6th Semester, the students shall present on the scheduled dates, their progress, twice to the Department committee based on which the Project sessional marks are awarded.

a.The project Seminar marks are awarded based on the project presentation and decomposition at the end of the semester by the Department committee.

b.Project final evaluation in the Term – end Exams, is based on project report and viva-voce.



# Eligibility to appear for the Term-end Exam.

Students, who have put in minimum of 75% attendances in each subject, shall be eligible to appear for the Term-end examination.

# Eligibility for Pass

A student shall be declared to have passed in a subject, if he/she secures at least 50% marks in the term- end examination, including internal assessment.

When a student reappears for the failed subject(s), the internal assessment marks originally secured by him/her in the first appearance in the subject(s), if any, will be caries forward.

A student shall be declared to have passed in a Core Bracket subject, if he/she secures at least a pass grade.

Promotion of the student to the next semester, is not automatic, but is dependent on certain other conditions.

# Classification of successful students

**7.1**On successful completion of the programme, the students will be classified as below: Distinction: Those securing aggregate marks of 75% and above in all the subjects.

First Class: Those securing aggregate marks of less than 75% but above 60% above in all the subjects; and Second Class: Those securing aggregate marks of less than 60% but above 50% above in all the subjects. **7.2Ranks:** Only students, who have passed each of the semester examination at the first appearance shall be eligible for award of Ranks. The First three ranks shall be notified.

# Award of Qualification

**Students will be awarded the MCA degree upon Fulfilment of the following criteria**

* 1. Must have passed all the subjects of the six semesters with minimum of 45% in each subject including internal assessment.
  2. Must have secured at least a pass grade in all the Core Bracket subjects.
  3. Must have secured a minimum of 45% marks in the project work (wherever applicable).
  4. Must have complied with all other assessment guidelines and criteria notified during the conduct of the programme.

# Maximum period for completion of the programme

The maximum period for the completion of the programme shall be five years from the date of joining the programme.

# General Guidelines Academic Integrity and Ethics

1. A student who has committed an act of academic dishonesty will be deemed to have failed to meet a basic requirement of satisfactory academic performance. Thus, academic dishonesty is not only a basis for disciplinary action but also is relevant to the evaluation of student’s level of performance and progress.



1. Where there has been violation of the basic ethos and principles of academic integrity and ethics, the Director/ Board of Examiners/ Course coordinators may use their discretion in terms of disciplinary action to be taken.
2. Academic dishonesty includes, but is not necessarily limited, to the following –
3. Cheating or knowingly assisting another student in committing an act of cheating.
4. Unauthorized possession of Examination materials, destruction or hiding of relevant materials.
5. Act of plagiarism.
6. Unauthorized changing of marks or making on examination records.

# Attendance

1. Students are required to attend and participate in all scheduled class sessions, guest lectures, workshops, outbound learning programs and club/ forum activities of both academic and non- academic nature.
2. Students may be dropped from the programs due to excessive and non-intimated absences.
3. Students must notify the program coordinator in writing, the reasons for absence, if any, from class sessions, activities and assessment components.
4. On notification of absences (including anticipated absences), the Director/ Programme coordinator would determine whether the absences could be rectified or whether it is possible to satisfactory complete the subject with the number of identified absences.

# General

1. The students are expected to spend a considerable amount of time in research, reading and practice.
2. All students are expected to develop and maintain a positive professional attitude and approach throughout the professional attitude and approach throughout the programme and in conduct of all other activities.
3. Attendance alone is not sufficient. Students are expected to participate, to help the class learn and understand the topics under consideration.
4. Food and drinks are not permitted in the classroom/ conference hall.
5. All students are expected to dress as per stipulated **dress** code.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Teaching Evaluation Scheme | | | | | | | | | |
| Course: Master of Computer Application (MCA) | Sem: I | Course Matrix With Effective From June 2023 | | | | | | | | | |
| Subject | | | Teaching Scheme | | | | Evaluation Scheme | | Total |
| Marks |
| Category | Name | Code | Hours | | | Credits | Theory | |  |
| Theory | Tutorial | Practical |  | Internal | University |
| Compulsory Subjects | |  |  |  |  |  |  | |
| Core Major | Object Oriented Programming Concepts | MCA 101 | 2 | 1 | 0 | 3 | 30 | 70 | 100 |
| Core Major | Database Technology | MCA 103 | 2 | 1 | 0 | 3 | 30 | 70 | 100 |
| Minor Elective | Data Structure Concepts | MCA 105 | 2 | 1 | 0 | 3 | 30 | 70 | 100 |
| MOOC /  Self Learning | Blockchain Technology / Prompt Engineering | MCA 108 / MCA10  9 | 0 | 3 | 0 | 3 | 30 | 70 | 100 |
| Inter - disciplinary | Mathematical Foundation for Computer Application | MCA 107 | 3 | 1 | 0 | 4 | 30 | 70 | 100 |
| Skill Enhancem ent Course | Object Oriented Programming Concepts Lab | MCA 102 | 0 | 0 | 4 | 2 | 15 | 35 | 50 |
| Skill  Enhancem ent Course | Database Technology Lab | MCA 104 | 0 | 0 | 4 | 2 | 15 | 35 | 50 |
| Skill  Enhancem ent Course | Data Structure Concepts Lab | MCA 106 | 0 | 0 | 4 | 2 | 15 | 35 | 50 |
| Ability Enhancem  ent Course | Soft Skills |  | 1 | 1 | 0 | 2 | 15 | 35 | 50 |
|  |  |  | **10** | **8** | **12** | **24** | **210** | **490** | **700** |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Teaching Evaluation Scheme | | | | | | | | | |
| **Course: Master of Computer Application (MCA) | Sem: II | Course Matrix With Effective From June 2023** | | | | | | | | | |
| **Courses** | | | **Teaching Scheme** | | | | **Evaluation Scheme** | | **Total** |
| **Marks** |
| **Category** | **Name** | **Code** | **Hours** | | | **Credit s** | **Theory** | |  |
| Theor  y | Tutori  al | Practical |  | Internal | University |
| Core Major | Advanced JAVA  Programming | MCA  201 | 3 | 1 | 0 | 4 | 30 | 70 | 100 |
| Core Major | Operating System Concepts and Shell  Programming | MCA 203 | 3 | 1 | 0 | 4 | 30 | 70 | 100 |
| Discipline Specific  Elective | Elective I |  | 3 | 1 | 0 | 4 | 30 | 70 | 100 |
| Elective I Lab |  | 0 | 0 | 4 | 2 | 15 | 35 | 50 |
| Discipline Specific  Elective | Elective II |  | 3 | 1 | 0 | 4 | 30 | 70 | 100 |
| Skill Enhancemen  t Course | Advanced JAVA Programming Lab | MCA 202 | 0 | 0 | 4 | 2 | 15 | 35 | 50 |
| Skill Enhancemen t Course | Operating System Concepts and Shell  Programming Lab | MCA 204 | 0 | 0 | 4 | 2 | 15 | 35 | 50 |
| Project Work | Industrial Training  with Project | MCA  205 | 0 | 4 | 4 | 4 | 100 | 0 | 100 |
|  |  |  | **12** | **8** | **16** | **26** | **265** | **385** | **650** |

# Elective Courses

|  |  |  |
| --- | --- | --- |
| Category | Name | Code |
| Elective I | Data Warehousing and  Data Mining | MCA 206 |
| Data Warehousing and  Data Mining Lab | MCA 207 |
| Elective I | Software Testing | MCA  208 |
| Software Testing  Lab | MCA  209 |
| Elective II | Advanced Networking  Concepts | MCA 210 |
| Elective II | Digital Image  Processing | MCA  211 |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Teaching Evaluation Scheme | | | | | | | | | |
| **Course: Master of Computer Application (MCA) | Sem: III | Course Matrix With Effective From June 2023** | | | | | | | | | |
| Courses | | | Teaching Scheme | | | | Evaluation Scheme | | Total |
| Mark  s |
| Category | Name | Code | Hours | | | Credits | Theory | |  |
| Theor  y | Tutorial | Practi  cal |  | Inte  rnal | Univer  sity |  |
| Core Major | Software Project  Management | MCA 301 | 3 | 1 | 0 | 4 | 30 | 70 | 100 |
| Discipline Specific Elective: **Machine Learning and AI** | Elective I: Machine  Learning | MCA  302 | 2 | 1 | 0 | 3 | 30 | 70 | 100 |
| Elective I Lab: Machine  Learning Lab | MCA 303 | 0 | 0 | 4 | 2 | 15 | 35 | 50 |
| Elective II: Artificial  Intelligence | MCA  304 | 2 | 1 | 0 | 3 | 30 | 70 | 100 |
| Elective II Lab: Artificial  Intelligence Lab | MCA  305 | 0 | 0 | 4 | 2 | 15 | 35 | 50 |
| Discipline Specific Elective: **Web Technology** | Elective I: Responsive and User Experience  (UX) Web Design | MCA 306 | 2 | 1 | 0 | 3 | 30 | 70 | 100 |
| Elective I Lab:  Responsive and User Experience (UX) We Lab | MCA 307 | 0 | 0 | 4 | 2 | 15 | 35 | 50 |
| Elective II: Advanced Web Development (Laravel, React,  Angular) | MCA 308 | 2 | 1 | 0 | 3 | 30 | 70 | 100 |
| Elective II Lab:  Advanced Web Development Lab | MCA 309 | 0 | 0 | 4 | 2 | 15 | 35 | 50 |
| Skill Enhancemen  t Course | IoT with Node MCU, Raspberry Pi) / Mobile  Application | MCA 310 /  MCA3 | 0 | 0 | 4 | 2 | 50 | 0 | 50 |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Development with  Flutter / Blockchain with Solidity | 11 /  MCA 312 |  |  |  |  |  |  |  |
| Project Work | Industrial Training with  Project | MCA  313 | 0 | 4 | 4 | 4 | 100 | 0 | 100 |
|  |  |  | **9** | **8** | **20** | **25** | **315** | **385** | **700** |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Teaching Evaluation Scheme | | | | | | | | | |
| **Course: Master of Computer Application (MCA) | Sem: IV | Course Matrix With Effective From June**  **2023** | | | | | | | | | |
| Courses | | | Teaching Scheme | | | | Evaluation Scheme | | Total |
| Mark  s |
| Category | Name | Code | Hours | | | Credit s | Theory | |  |
| Theor  y | Tutori  al | Practic  al |  | Intern  al | Universit  y |
| Internship | Internship | MCA 401 |  |  |  | 20 | 150 | 350 | 500 |
| Ability Enhancement Course | Organizational Behaviour and Professional  Ethics | MCA 402 |  |  |  | 4 | 100 | 0 | 100 |
|  |  |  |  |  |  | **24** | **250** | **350** | **600** |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **SYLLABUS**  **PROGRAM: MCA SEMESTER: I WEF: 2024-25** | | | | | | |
| **Course Code: MCA101** | | | **Credit: 04** | **Course: Object Oriented Programming**  **Concepts** | **L: 03 | T: 01 | P: 00** | |
| Prerequisites: | The prerequisites for Object-Oriented Programming include a basic understanding of programming concepts like variables, data types, control structures, functions, algorithms, data structures, and familiarity with a programming language. | | | | | |
| Objectives: | The objective of this course is to make student understand and implement concepts of Object-oriented programming, which is in itself one of the most powerful programming languages, it aims to implement real-world entities like inheritance, hiding, polymorphism,  etc. in programming. | | | | | |
| Course Outcome: | Upon successfully finishing the course, students will have the capability to: | | | | | |
|  | **No.** | **Course Outcome** | | | | **BT Level** |
| CO1 | Understand the basic building block of programming. | | | | Understand |
| CO2 | Apply the concept of class and object, memory allocation and deallocation concepts. | | | | Understand,  Apply |
| CO3 | Gain the understanding of operator overloading and will also apply inheritance using real world scenarios. | | | | Apply  Analysis |
| CO4 | Implement the concept of pointer, need of friend function and one the most important feature polymorphism. | | | | Understand  Apply |
| CO5 | Analyze the process of data storing and accessing through file stream classes. | | | | Apply |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Program |  | | | | | | | | | | | | | |
| Outcomes and Course Outcomes Mapping: |  | **Course Outcomes** | **Program Outcomes** | | | | | | | | | | | |
| PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO2 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO3 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO4 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO5 |  |  |  |  |  |  |  |  |  |  |  |  |
|  | | | | | | | | | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Syllabus** | | | | | |
| **No.** | **Module Description** | | | **BT Level** | **Hours** |
| **1** | **Programming Paradigms** | | | **U, AP, AN** | **15** |
|  | 1.1 | Programming Language Concepts | | U | 2 |
|  | 1.2 | Types Of Programming Language and Its Application Area | | U,AP | 1 |
|  | 1.3 | Source File Creation, Compilation and Linking | | U,AP | 2 |
|  | 1.4 | Features And Structure of C++ Program | | U,AP,AN | 1 |
|  | 1.5 | Data Types, Tokens: Keywords, Identifiers, Variables and Constants, Operators | | U,AP,AN | 2 |
|  | 1.6 | Control Statements: Branching, Looping and Jumping | | U, AP, AN | 2 |
|  | 1.7 | Array Declaration and Initialization | | U,AP | 2 |
|  | 1.8 | Access the Array Elements | | U,AP | 1 |
|  | 1.9 | Types of Arrays | | U,AP | 2 |
| **2** | **Class, Object, Constructor and Destructor** | | | **U, AP, AN** | **11** |
|  | 2.1 | Object Oriented Programming Concepts, Advantage | | U,AP | 2 |
|  | 2.2 | Object and Class | | U,AP | 1 |
|  | 2.3 | Member Function | | U,AP,AN | 1 |
|  | 2.4 | Array within the Class | | U,AP,AN | 1 |
|  | 2.5 | Memory Allocation of Objects | | **U, AP, AN** | 1 |
|  | 2.6 | Friend Function | | U,AP | 1 |
|  | 2.7 | Local Class | | U,AP | 1 |
|  | 2.8 | Constructors: Parameterized, Multiple, Default Argument | | U,AP | 1 |
|  | 2.9 | Dynamic Initialization of Objects, Copy Constructor, Dynamic Constructor | | U,AP | 1 |
|  | 2.10 | Destructors | | U,AP,AN | 1 |
| **3** | **Operator Overloading and Inheritance** | | | **U, AP, AN** | **12** |
|  | 3.1. | Operator Overloading: Unary And Binary | | U,AP | 2 |
|  | 3.2. | Overloading Binary Operators Using Friends | | U,AP | 2 |
|  | 3.3. | Rules of Overloading Operators, Type Conversion | | U,AP,AN | 2 |
|  | 3.4. | Inheritance, Derived Classes | | U,AP,AN | 2 |
|  | 3.5 | Inheritance: Single, Multilevel, Multiple, Hierarchical and Hybrid | | U,AP | 1 |
|  | 3.6 | Virtual Base Classes, Abstract Class | | U,AP | 1 |
|  | 3.7 | Constructors In Derived Classes, Member Classes | | U,AP | 2 |
| **4** | **Pointer, Virtual Function and Polymorphism** | | | **U, AP, AN** | **10** |
|  | 4.1. | Pointers: Pointers To Objects, This Pointer | | U,AP | 2 |
|  | 4.2. | Pointer To Derived Classes | | U,AP | 2 |
|  | 4.3. | Virtual Function, Pure Virtual Function | | U,AP | 2 |
|  | 4.4. | Polymorphism: Compile Time, Run Time | | U,AP | 2 |
|  | 4.5. | Overloading and overriding | | U,AP | 2 |
| **5** | **Console I/O Operations and File Handling** | | | **U, AP, AN** | **12** |
|  | 5.1. | Stream Classes | | U,AP | 2 |
|  | 5.2. | I/O Operations: Unformatted and Formatted | | U,AP | 2 |
|  | 5.3. | Managing Output with Manipulators | | U,AP | 1 |
|  | 5.4. | Classes For File Stream Operations | | U,AP | 1 |
|  | 5.5. | Opening and Closing a File, Detecting End-of-File | | U,AP | 1 |
|  | 5.6 | File Modes, File Pointers and Their Manipulations | | U,AP | 1 |
|  | 5.7 | Sequential Input and Output Operations | | U,AP | 1 |
|  | 5.8 | Random Access File | | U,AP | 1 |
|  | 5.9 | Error Handling During File Operations | | U,AP | 2 |
|  |  |  | |  |  |
| Course Modules and Course Outcomes Mapping: | | | |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **#** | **Module** | **Course Outcomes** | | | | | |  | | **CO1** | **CO2** | **CO3** | **CO4** | **CO5** | | 1 | Programming Paradigms |  |  |  |  |  | | 2 | Class, Object, Constructor and Destructor |  |  |  |  |  | | 3 | Operator Overloading and Inheritance |  |  |  |  |  | | 4 | Pointer, Virtual Function and Polymorphism |  |  |  |  |  | | 5 | Console I/O Operations and File Handling |  |  |  |  |  | | | |
| Textbooks/  Resources: | | | 1. E. Balagurusamy , “Object Oriented Programming with C++”, Tata McGraw-Hill: **TB#1**  2. Object-Oriented Programming in C++" by Robert Lafore  3. "Head First Object-Oriented Analysis and Design" by Brett D. McLaughlin | | |
| Reference Books/  Resources | | | 1. Herbert Schildt , “The Complete Reference” Tata McGraw-Hill  2.Robert Lafore , “Object Oriented Programming in Turbo C++” The Waite Group  3.Programming in Modern C++ NPTEL SWAYAM: <https://onlinecourses.nptel.ac.in/noc23_cs78/preview>  4.C++ Tutorial <https://www.javatpoint.com/cpp-tutorial> | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **SYLLABUS**  **PROGRAM: MCA SEMESTER: I WEF: 2023-24** | | | | | | |
| **Course Code: MCA103** | | | **Credit: 04** | **Course: Database Technology** | **L: 03 | T: 01 | P: 00** | |
| Prerequisites: | Basic knowledge of Database Management. | | | | | |
| Objectives: | Familiarize students with the fundamental principles of database systems, process of designing relational database schemas, transaction processing. Introduce the concept of object-oriented databases and provide hands-on experience in SQL & procedural SQL programming. | | | | | |
| Course Outcome: | Upon successfully finishing the course, students will have the capability to: | | | | | |
|  | **No.** | **Course Outcome** | | | | **BT Level** |
| CO1 | Gain an understanding of database concepts, data models, database architecture, and user roles. | | | | Understand |
| CO2 | Explore the process of designing efficient relational database schemas for applications. | | | | Apply |
| CO3 | Demonstrate the application of conditional and iterative statements, and develop user-defined functions, stored procedures, and triggers using procedural SQL. | | | | Apply |
| CO4 | Apply the concepts of transactions, understand their states, and grasp the concepts of concurrency and the necessity of serializability. | | | | Apply |
| CO5 | Analyze the principles of Object-Oriented databases. | | | | Analyze |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Program | | |  | | | | | | | | | | | | | | |
| Outcomes and Course Outcomes Mapping: | | | **Course Outcomes** | **Program Outcomes** | | | | | | | | | | | | | |
| PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | | PO10 | PO11 | | PO12 |
| CO1 |  |  |  |  |  |  |  |  |  | |  |  | |  |
| CO2 |  |  |  |  |  |  |  |  |  | |  |  | |  |
| CO3 |  |  |  |  |  |  |  |  |  | |  |  | |  |
| CO4 |  |  |  |  |  |  |  |  |  | |  |  | |  |
| CO5 |  |  |  |  |  |  |  |  |  | |  |  | |  |
|  | | | | | | | | | | | | | | |
| **Syllabus** | | | | | | | | | | | | | | | | | |
| **No.** | **Module Description** | | | | | | | | | | | | **BT Level** | | | **Hours** | |
| **1** | **Introduction to Database Management System** | | | | | | | | | | | | **U, AP, AN** | | | **10** | |
|  | 1.1 | Purpose of Database Systems | | | | | | | | | | | U,AP | | | 2 | |
|  | 1.2 | View of Data: Data Abstraction, Instances and Schemas | | | | | | | | | | | U,AP | | | 1 | |
|  | 1.3 | Data Models: Relational Model, Entity-Relationship Model, Object-Based Data Model, Semistructured Data Model, Database Languages | | | | | | | | | | | U,AP,AN | | | 3 | |
|  | 1.4 | Data Storage and Querying, Database Architecture | | | | | | | | | | | U,AP,AN | | | 2 | |
|  | 1.5 | Database Users and Administrators | | | | | | | | | | | U,AP | | | 2 | |
| **2** | **Relational Data Modeling and Database Design** | | | | | | | | | | | | **U, AP, AN** | | | **10** | |
|  | 2.1 | Relational Model Concepts, Super Key, Candidate Key and Primary Key | | | | | | | | | | | U,AP | | | 2 | |
|  | 2.2 | Constraints: Domain, Key, Entity and Referential Integrity constraints | | | | | | | | | | | U,AP | | | 1 | |
|  | 2.3 | E.F. Codd’s Rule | | | | | | | | | | | U,AP,AN | | | 2 | |
|  | 2.4 | Functional dependency, Armstrong’s Inference rules | | | | | | | | | | | U,AP,AN | | | 1 | |
|  | 2.5 | Decomposition of Relations: Lossless Join and Dependency Preservation property | | | | | | | | | | | U,AP | | | 2 | |
|  | 2.6 | Normalization: First, Second, Third, BCNF, PJNF | | | | | | | | | | | U,AP | | | 2 | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **3** | **SQL and Procedural SQL** | | **U, AP, AN** | **8** |
|  | 3.1. | Conditional statements and Iterative statements | U,AP | 2 |
|  | 3.2. | User-defined functions | U,AP | 2 |
|  | 3.3. | Stored Procedures, Parameter types: IN, OUT and INOUT | AP | 2 |
|  | 3.4. | Triggers: Introduction, Needs, Before trigger and After trigger | APS | 2 |
| **4** | **Transaction management and Concurrency** | | **U, AP, AN** | **12** |
|  | 4.1. | Transaction: Introduction, Transaction Model | U,AP | 2 |
|  | 4.2. | Properties of Transactions | U,AP | 2 |
|  | 4.3. | Transaction isolation, Schedules: Serial, Non-Serial Schedules | U,AP,AN | 2 |
|  | 4.4. | Serializability, Conflict Serializability | U,AP,AN | 2 |
|  | 4.5. | Concurrency Control | U,AP | 2 |
|  | 4.6. | Concurrency Control Protocols: Lock based and Timestamp based | U,AP | 2 |
|  | 4.7. | Deadlock Handling: Detection and Prevention | U,AP |  |
| **5** | **Object-Oriented Database** | | **U, AP, AN** | **10** |
|  | 5.1. | Limitations of RDBMS | Understand | 2 |
|  | 5.2. | Introduction: OODBMS and ORDBMS | Understand | 2 |
|  | 5.3. | Storing and Accessing Objects in a Relational Database | Understand | 2 |
|  | 5.4. | Object-Oriented Database Design | Understand | 2 |
|  | 5.5. | Introduction to Object-Oriented Data Models | Understand | 2 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **#** | **Module** | **Course Outcomes** | | | | |
|  | | **CO1** | **CO2** | **CO3** | **CO4** | **CO5** |
| 1 | Introduction to Database Management  System |  |  |  |  |  |
| 2 | Relational Data Modeling and Database Design |  |  |  |  |  |
| 3 | Working with Procedural SQL |  |  |  |  |  |
| 4 | Transaction management and  Concurrency control |  |  |  |  |  |
| 5 | Object-Oriented Database |  |  |  |  |  |

|  |  |
| --- | --- |
| Course Modules and Course Outcomes Mapping: |  |
| Textbooks/  Resources: | 1.Henry F. Korth, “Database System Concepts”, Tata McGraw Hills 2.Ivan Bayross, “MySQL 5.1 for Professionals”, SPD |
| Reference Books/  Resources | 1.Elmasri and Navathe, “Fundamentals of Database Systems”, Pearson Education. 2.Thomas Connolly and Carolyn Begg, “Database Systems, A Practical Approach to Design Implementation and Management", Pearson Education   1. MySQL Reference Manual - https://dev.mysql.com/doc/refman/8.0/en/ 2. PostgreSQL Reference Manual - https://[www.postgresql.org/docs/](http://www.postgresql.org/docs/) |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **SYLLABUS**  **PROGRAM: MCA SEMESTER: I WEF: 2023-24** | | | | | | |
| **Course Code: MCA 106** | | | **Credit: 04** | **Course: Data Structure Concepts** | **L: 03 | T: 01 | P: 00** | |
| Prerequisites: | Nil | | | | | |
| Objectives: | To provide the knowledge of basic data structures and their implementations. Also understand the importance of data structures in context of writing efficient programs and develop skills to apply appropriate data structures in problem solving. | | | | | |
| Course Outcome: | Upon successfully finishing the course, students will have the capability to:S | | | | | |
|  | **No.** | **Course Outcome** | | | | **Level** |
| CO1 | Student gets able to understand and Implement linear and non- linear data structures. | | | | Understanding |
| CO2 | Student gets ability to use structures such as stack, queues, linked list, trees and graphs. | | | | Understanding  Apply |
| CO3 | Analyze and implement various searching and sorting techniques. | | | | Understanding  Apply |
| CO4 | Student gets able to understand and Implement concept of tree and graph. | | | | Understanding  Apply |
| CO5 | Evaluate algorithms and data structures in terms of time and space complexity of basic operations. | | | | Understanding  Analysis Evaluation |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Program |  | | | | | | | | | | | | | |
| Outcomes and Course Outcomes Mapping: |  | **Course Outcomes** | **Program Outcomes** | | | | | | | | | | | |
| PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO2 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO3 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO4 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO5 |  |  |  |  |  |  |  |  |  |  |  |  |
|  | | | | | | | | | | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Module Description** | | **BT Level** | **Hours** |
| **1** | **Linear Data Structure** | | **U, AP, AN** | **14** |
|  | 1.1 | Data structure concepts , Data type, Abstract data type | U | 2 |
|  | 1.2 | Linear data structures using sequential organization, Operations | U,AP | 2 |
|  | 1.3 | Linear Array in data structure and its classification, Properties and  representation of an array , Operation and Memory location | U,AP | 2 |
|  | 1.4 | Searching Algorithms: Linear, Binary | U,AP,AN | 4 |
|  | 1.5 | Sorting Algorithm—Insertion, Selection, Merge sort | U,AP,AN | 4 |
| **2** | **Stack, Queue and Recursion** | | **U, AP, AN** | **10** |
|  | 2.1 | Representation of Stacks using sequential organization, Applications | U,AP | 5 |
|  | 2.2 | Recursion and its applications | U,AP | 5 |
|  | 2.3 | Queue, Representation of Queues using sequential organization, Dequeue,  Priority Queue | U,AP |  |
| **3** | **Linked List** | | **U, AP, AN** | **12** |
|  | 3.1 | Linked list and its representation | U | 2 |
|  | 3.2 | Operations on Linked list: Traversing, Searching, Insertion, Deletion | U,AP | 5 |
|  | 3.3 | Memory Allocation | U,AP | 5 |
| **4** | **Tree and Graph** | | **U, AP, AN** | **12** |
|  | 4.1 | Tree concepts | U,AP | 1 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **#** | **Module** | **Course Outcomes** | | | | |
|  | | **CO1** | **CO2** | **CO3** | **CO4** | **CO5** |
| 1 | Linear Data Structure |  |  |  |  |  |
| 2 | Stack, Queue and Recursion |  |  |  |  |  |
| 3 | Linked List |  |  |  |  |  |
| 4 | Tree and Graph |  |  |  |  |  |
| 5 | Algorithm Analysis and Design |  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 4.2 | Binary Tree-Representation | U,AP | 1 |
|  | 4.3 | Operations: Searching, Insertion, Deletion | U,AP | 3 |
|  | 4.4 | Algorithms : Binary Search Tree and AVL | U,AP | 4 |
|  | 4.5 | Graph, Graph Representation , Operations : Searching, Insertion, Deletion,  Traversing | U,AP | 3 |
| **5** | **Algorithm Analysis and Design** | | **U, AP, AN** | **12** |
|  | 5.1 | The Role of Algorithm in Computing, Characteristics of algorithm, P and NP  problem | U,AN | 2 |
|  | 5.2 | Analyzing algorithms: Time and space complexity, Execution time | U,AN,EV | 2 |
|  | 5.3 | Asymptotic notations | U,AN,EV | 3 |
|  | 5.4 | Algorithm design techniques: Greedy, Divide and conquer, Dynamic  programming (one example of each) | U,AN,EV | 5 |

|  |  |
| --- | --- |
| Course Modules and Course Outcomes Mapping: |  |
| Text Books/ Resources: | 1. Seymour Lipschutz, ”Data Structures”, McGraw Hill Education: TB#1 2. Alfred V. Aho, John E. Hopcroft, Jeffry D. Ullaman” Data Structure and Algorithms”, Pearson Education: TB#2 3. Thomas H Coreman , “Introduction to algorithms”, MIT Press: TB#3 |
| Reference Books/  Resources | 1. Michael T. Goodrich, Roberto Tamassia, David M. Mount, “Data Structure and Algorithm”, John Wiley & Sons |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **SYLLABUS**  **PROGRAM: MCA SEMESTER: I WEF: 2023-24** | | | | | | |
| **Course Code: MCA107** | | | **Credit: 04** | **Course: Mathematical Foundation of Computer Application** | | **L: 03 | T: 01 | P: 00** |
| Prerequisites: | Nil | | | | | |
| Objectives: | Familiarize students with the fundamental principles of set theory, function. Introduce the concept of logical connectivity, basic concept of Boolean algebra. Acquire knowledge on lattice and coding theory. Understand group theory and basic concepts of graph theory. | | | | | |
| Course Outcome: | Upon successfully finishing the course, students will have the capability to: | | | | | |
|  | **No.** | **Course Outcome** | | | **BT Level** | |
| CO1 | Understand sets and of relations, Identify different types relations, functions, compute composition and inverse of a function. | | | Understand, Apply, Analyse | |
| CO2 | Gain knowledge on lattice and coding theory. | | | Understand, Apply | |
| CO3 | Gain knowledge on logical connectivities, basic concept of Boolean algebra and its application in logic building. | | | Understand, Apply, Analyse | |
| CO4 | Able to define and explain the basic concepts of graph theory and its applications. | | | Understand, Apply, Analyse | |
| CO5 | Describe and understand group theory which is crucial ingredients in computer science | | | Understand, Apply | |
| Program Outcome: | Upon successfully finishing the program, students shall be able to: | | | | | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Program |  | | | | | | | | | | | | |
| Outcomes and Course Outcomes Mapping: | **Course Outcomes** | **Program Outcomes** | | | | | | | | | | | |
| PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO2 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO3 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO4 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO5 |  |  |  |  |  |  |  |  |  |  |  |  |
|  | | | | | | | | | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Curriculum** | | | | |
| **No.** | **Module Description** | | **BT Level** | **Hours** |
| **1** | **Set Theory, Mathematical Logic, Relation and Function** | | **U, AP, AN** | **10** |
|  | 1.1 | Introduction to Set theory, Cartesian Product | U | 2 |
|  | 1.2 | Statements and Notations, Logical Connectives | U, AP | 2 |
|  | 1.3 | Tautology, Contradiction, Logical equivalence | U, AP | 2 |
|  | 1.4 | Relation, Types of Binary relation, Equivalence relation | U, AP | 2 |
|  | 1.5 | Function, Properties of function, Composition of Functions | U, AP, AN | 2 |
| **2** | **POSETS and Lattices** | | **U, AP** | **10** |
|  | 2.1 | Introduction, Partial order relation | U | 2 |
|  | 2.2 | Partial ordered set, HASSE diagram | U, AP | 3 |
|  | 2.3. | Lattice, Sub-Lattices, Well ordered set, Complete lattice | U, AP | 3 |
|  | 2.4 | Distributive and Complemented lattice | U | 2 |
| **3** | **Boolean Algebra** | | **U, AP, AN** | **10** |
|  | 3.1 | Basic concepts of Boolean Algebra, Boolean lattice, Boolean algebra | U | 2 |
|  | 3.2 | Boolean functions, Disjunctive and Conjunctive normal form, Complement function, Bool’s Expansion theorem | U, AP | 3 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **#** | **Module** | **Course Outcomes** | | | | |
|  | | **CO1** | **CO2** | **CO3** | **CO4** | **CO5** |
| 1 | Set Theory, Mathematical Logic,  Relation and Function |  |  |  |  |  |
| 2 | POSETS and Lattices |  |  |  |  |  |
| 3 | Boolean Algebra |  |  |  |  |  |
| 4 | Graph Theory |  |  |  |  |  |
| 5 | Semi Group and Monoids |  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 3.3 | Karnaugh map method for simplification of Boolean expressions | U, AP, AN | 3 |
|  | 3.4 | Applications of Boolean Algebra in switching circuits, logic circuits | U, AP, AN | 2 |
| **4** | **Graph Theory** | | **U, AP, AN** | **12** |
|  | 4.1 | Basic concepts of graph theory | U | 2 |
|  | 4.2 | Subgraph, Walk, Paths and Circuits | U | 2 |
|  | 4.3 | Types of graphs | U | 3 |
|  | 4.4 | Matrix Representation of Graphs, Directed Graphs | U,AP | 2 |
|  | 4.5 | Tree and its properties, Rooted Tree, Binary Trees, Spanning tree, Fundamental Circuits | U, AP, AN | 3 |
| **5** | **Semi Group and Monoids** | | **U, AP** | **8** |
|  | 5.1 | Algebraic structure, Binary operation, Properties, Semi group, Monoid, Group theory, | U | 2 |
|  | 5.2 | Abelian group, Cyclic group, Generators, Permutation group, Subgroup, | U, AP | 2 |
|  | 5.3 | Homomorphism, Isomorphism and Automorphism. | U, AP | 2 |
|  | 5.4 | Cosets, Langranges Theorem, Normal Subgroup and Quotient group | U, AP | 2 |

|  |  |
| --- | --- |
| Course Modules and Course Outcomes Mapping: |  |
| Text Books/  Resources: | 1.H.K Pathak, J.P. Chauhan “Advanced Discrete Mathematics”, Shree Shiksha Sahitya Prakashan. |

|  |  |
| --- | --- |
|  | 2.Swapan Kumar Sarkar, “Discrete Mathematics”, S. Chand & Company Ltd. |
| Reference Books/  Resources | 1.Ralph, Gimaldi, Discrete and Combinatorial Mathematics, Pearson Education. 2.N.Deo, Graph Theory with Applications to Engineering and Computer Science, Prentice Hall.  3.H.K Pathak, Discrete Mathematics  4.Kolman, B, Busby, R.C.Ross, S.C. Discrete Mathematical Structures, Pearson Education. 5.Liu, C.L. Elements of Discrete Mathematics, Tata McGraw Hill. |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **SYLLABUS**  **PROGRAM: MCA SEMESTER: II WEF: 2023-24** | | | | | |
| **Course Code: MCA201** | | | **Credit: 04** | **Course: Advanced JAVA Programming** | **L: 03 | T: 01 | P: 00** |
| Prerequisites: | The prerequisites for Advanced Java Programming include a strong understanding of core Java concepts such as object-oriented programming (OOP), collections, exception handling, multithreading, Java 8 features (lambda expressions, streams), JVM architecture, file I/O, and basic knowledge of databases and SQL. | | | | |
| Objectives: | This course aims to equip students with a comprehensive understanding of Java programming and related technologies. By the end of the course, students will achieve proficiency in Object-Oriented Programming (OOP) concepts, including class, object,  encapsulation, polymorphism, inheritance, and abstraction. | | | | |
| Course Outcome: | Upon successfully finishing the program, students shall be able to: Develop Business Logic, Develop GUI & Web Application | | | | |
|  | **No.** | **Course Outcome** | | | **BT Level** |
| CO1 | Student will understand the basic building block of core Java programming. | | | Understand, Apply, Analyze |
| CO2 | Students should gain a comprehensive understanding of Java FX and its various features, enabling them to create interactive and visually appealing graphical user interfaces. | | | Understand, Apply, Analyze |
| CO3 | Students should gain a solid understanding of J2EE concepts and be able to develop Servlets that handle client requests, manage form data, handle cookies, and implement session tracking in web applications. | | | Understand, Apply, Analyze |
| CO4 | Students should gain a comprehensive understanding of JSP concepts and be able to develop dynamic web pages using scripting elements, implicit objects, directive elements, and action elements. | | | Understand, Apply, Analyze |
| CO5 | Students should gain a comprehensive understanding of Spring and Spring Boot concepts and be able to develop applications using these frameworks. | | | Understand, Apply, Analyze |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Program Outcome: | Upon successfully finishing the program, students shall be able to: | | | | | | | | | | | | | |
| Program |  | | | | | | | | | | | | | |
| Outcomes | **Course** | **Program Outcomes** | | | | | | | | | | | | |
| and | **Outcomes** |
| PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | | PO12 |
| Course |  |
| CO1 |  |  |  |  |  |  |  |  |  |  | |  |  |
| Outcomes |
| Mapping: | CO2 |  |  |  |  |  |  |  |  |  |  | |  |  |
|  | CO3 |  |  |  |  |  |  |  |  |  |  | |  |  |
|  | CO4 |  |  |  |  |  |  |  |  |  |  | |  |  |
|  | CO5 |  |  |  |  |  |  |  |  |  |  | |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Module Description** | | BT Level | Hours |
| 1 | **Object Oriented Programming Concepts and Implementations** | | **U, AP** | **12** |
|  | 1.1 | OOPS Concepts and implementation: Class, Object, Encapsulation,  Polymorphism, Inheritance, Abstraction | U, AP | 2 |
|  | 1.2 | Package Concepts and Implementation: Defining and use of package | U, AP | 2 |
|  | 1.3 | Managing Errors and Exceptions | U, AP | 2 |
|  | 1.4 | Multithreading: Concepts, Life Cycle, Methods | U, AP | 2 |
|  | 1.5 | Network Programming: Introduction, Classes and its methods | U, AP | 2 |
|  | 1.6 | Java Database Connectivity: Architecture, Drivers, Connectivity with  database | U, AP | 2 |
| 2 | **JAVA FX Technology** | | **U, AP** | **10** |
|  | 1.1 | Introduction to Java FX, Features, Architecture & Application | U | 1 |
|  | 1.2 | Java 2D Shapes, Colors, Text | U, AP | 1 |
|  | 1.3 | Java FX Effects | U, AP | 1 |
|  | 1.4 | Java FX Transformation | U, AP | 1 |
|  | 1.5 | Java FX Animation | U, AP | 1 |
|  | 1.6 | Java FX 3D Shapes | U, AP | 1 |
|  | 1.7 | Java FX Layout | U, AP | 1 |
|  | 1.8 | Java FX UI Controls | U, AP | 1 |
|  | 1.9 | Java FX Images | U, AP | 1 |
|  | 1.10 | Java FX Event Handling | U, AP | 1 |
| 3 | **Servlet Technology** | | **U, AP** | **10** |
|  | 3.1 | J2EE Introduction and Architecture | U | 1 |
|  | 3.2 | Java Servlet: Basic Servlet Structure | U | 2 |
|  | 3.3 | Servlet Life Cycle | U | 1 |
|  | 3.4 | Reading Form Data from Servlet | U, AP | 1 |
|  | 3.5 | Handlin the Client Request, Generating the Server Response | U, AP | 1 |
|  | 3.6 | Handling Cookies: Benefit of Cookies, Deleting Cookies, Sending and  Receiving Cookies, Cookies Attributes | U, AP | 2 |
|  | 3.7 | Session Tracking: Need of Session Tracking, Session Tracking Basics,  Session Tracking API, | U, AP | 2 |
| 4 | **JSP Technology** | | **U, AP** | **8** |
|  | 4.1 | Introduction, Need and Benefit of JSP, Life Cycle of JSP | U | 1 |
|  | 4.2 | JSP Scripting Elements : Scriptlet tag, Expression Tag, Declaration Tag | U, AP | 2 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 4.3 | Implicit Object: Request, Response, Config, Application, Session, Page  Context, Page, Exception | U, AP | 2 |
|  | 4.4 | Directive Elements: Page directive, include directive, taglib directive | U, AP | 2 |
|  | 4.5 | Action Elements: forward, include | U, AP | 1 |
| 5 | **Spring and Spring Boot Framework** | | **U, AP** | **14** |
|  | 5.1 | Introduction to Spring Initializing and Writing Spring application | U, AP | 1 |
|  | 5.2 | Dependency Injection: IOC Container, Dependency Injection,  Constructor Injection. | U, AP | 2 |
|  | 5.3 | Developing web applications: Displaying & processing information,  Validating Form Input, Working with View Controller | U, AP | 2 |
|  | 5.4 | Working with Data: Reading and Writing data with JDBC, Working with  Spring data JDBC | U, AP | 2 |
|  | 5.5 | Introduction to Spring Boot, Difference between Spring and Spring  Boot, Spring Boot Architecture | U | 1 |
|  | 5.6 | Project Component: Annotations, Dependency Management,  Application Properties, | U, AP | 2 |
|  | 5.7 | Developing Spring Boot Applications: Putting spring boot to work,  using starter dependencies, using automatic configurations | U, AP | 2 |
|  | 5.8 | AOP (Aspect Oriented Programming) : Before, After, Around,  Returning, Throwing | U, AP | 2 |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Course Modules and Course Outcomes Mapping: | |  |  |  |  | | | | |
| **#** | **Module** | **Course Outcomes** | | | | |
|  | | **CO1** | **CO2** | **CO3** | **CO4** | **CO5** |
| 1 | Object Oriented Programming Concepts and Implementations | √ |  |  |  |  |
| 2 | JAVA FX Technology |  | √ |  |  |  |
| 3 | Servlet Technology |  |  | √ |  |  |
| 4 | JSP Technology |  |  |  | √ |  |
| 5 | Spring and Spring Boot Framework |  |  |  |  | √ |
|  | | | | | | | |
| Text Books/ Resources: | 1. E. Balagurusamy , “Programming with Java”, Tata McGraw-Hill: TB#1 2. Marty Hall, Larry Brown, “Core Servlet and JavaServer Pages”, PEARSON: TB#2 | | | | | | | | |

|  |  |
| --- | --- |
|  | 1. Carl Dea, Mark Heckler, Gerrit Grunwald, Jose Pereda Ph.D, Sean M Philips, “Java FX 8 Introduction by Example”, Apress: TB#3 2. Craig Walls, “Spring IN ACTION”,MANNING: TB#4 3. Craig Walls, Andrew Glover, “Spring Boot IN ACTION”,MANNING: TB#5 |
| Reference Books/  Resources | 1. Bruce Eckel, “THINKING IN JAVA”, PEARSON 2. JDK Release Notes - https://[www.oracle.com/java/technologies/javase/jdk-](http://www.oracle.com/java/technologies/javase/jdk-) relnotes-index.html 3. JavaFX - https://jenkov.com/tutorials/javafx/index.html |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **SYLLABUS**  **PROGRAM: MCA SEMESTER: II WEF: 2023-24** | | | | | |
| **Course Code: MCA203** | | **Credit: 04** | **Course: OS CONCEPTS & SHELL PROGRAMMING** | **L: 03 | T: 01 | P: 00** | |
| Prerequisites: | Nil | | | | |
| Objectives: | To provide a basic understanding of the operating system's function, architecture, services, and interservice coordination. To learn the fundamentals of creating, running, and troubleshooting shell scripts. Discuss the fundamental control structures, variables, and data kinds that are needed to build scripts. | | | | |
| Course Outcome: | Upon successfully finishing the course, students will have the capability to: | | | | |
|  | **No.** | **Course Outcome** | | | **BT Level** |
| CO1 | Understanding the concept and Types of Operating Systems, Structure and Functions of OS, and Basics of System software’s. | | | Understand |
| CO2 | Understand the various process states and their transitions, and context switching in managing processes. Demonstrate the different CPU scheduling methods and Concurrency control. | | | Analysis |
| CO3 | Determine and understand the concept of memory management, swapping, paging, segmentation, virtual memory management and page replacement algorithms. | | | Understand |
| CO4 | Understand the disk architecture, various disk scheduling algorithms, file systems implement, access method selection, and file allocation methods. | | | Understand |
| CO5 | Analyze the various features and functionalities of different shells in shell programming, including control flow statements , functions, and file handling mechanism. | | | Analyze |
| Program Outcome: | Upon successfully finishing the program, students shall be able to: | | | | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Program Outcomes and Course Outcomes Mapping: | | |  | | | | | | | | | | | | | |
| **Course Outcomes** | **Program Outcomes** | | | | | | | | | | | | |
| PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | | PO12 |
| CO1 |  |  |  |  |  |  |  |  |  |  |  | |  |
| CO2 |  |  |  |  |  |  |  |  |  |  |  | |  |
| CO3 |  |  |  |  |  |  |  |  |  |  |  | |  |
| CO4 |  |  |  |  |  |  |  |  |  |  |  | |  |
| CO5 |  |  |  |  |  |  |  |  |  |  |  | |  |
|  | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | | | | |
| **No.** | **Module Description** | | | | | | | | | | | | **BT Level** | | **Hours** | |
| **1** | **Introduction to Operating System** | | | | | | | | | | | |  | | **10** | |
|  | 1.1 | Introduction to Operating System. | | | | | | | | | | | Understand | | 2 | |
|  | 1.2 | Need and Functions of Operating Systems | | | | | | | | | | | Understand | | 2 | |
|  | 1.3 | Computer System Operations | | | | | | | | | | | Understand | | 3 | |
|  | 1.4 | Types of OS -Batch processing, Multi-programming, Time sharing | | | | | | | | | | | Understand | | 2 | |
|  | 1.5 | Operating-System Services | | | | | | | | | | | Understand | | 2 | |
|  | 1.6 | System Calls. | | | | | | | | | | | Understand | | 2 | |
|  | 1.7 | Operating-System Structure | | | | | | | | | | | Understand | | 2 | |
|  | 1.8 | Design Goals | | | | | | | | | | | Understand | | 2 | |
| **2** | **Process Management and Synchronization** | | | | | | | | | | | |  | | **10** | |
|  | 2.1 | Process Concepts | | | | | | | | | | | Understand | | 2 | |
|  | 2.2 | Process State | | | | | | | | | | | Understand | | 1 | |
|  | 2.3 | Process Control Block | | | | | | | | | | | Understand | | 2 | |
|  | 2.4 | Operations on Processes | | | | | | | | | | | Understand | | 1 | |
|  | 2.5 | Inter process communications | | | | | | | | | | | Understand | | 1 | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 2.6 | Process Scheduling | Understand | 2 |
|  | 2.7 | CPU Scheduling algorithms | Apply | 3 |
|  | 2.8 | Process Threads - Overview | Understand | 1 |
|  | 2.9 | Critical Section problem | Understand | 2 |
|  | 2.10 | Semaphores. | Understand | 3 |
|  | 2.11 | Classical problems of synchronization | Understand | 2 |
|  | 2.12 | Deadlock Characterization | Understand | 2 |
|  | 2.13 | Deadlock Handling: Avoidance, Detection &Prevention | Apply | 3 |
| **3** | **Storage Management** | |  | **8** |
|  | 3.1. | Contiguous Memory Allocation | Understand | 2 |
|  | 3.2. | Paging Techniques: Swapping, Paging, Segmentation, Fragmentation | Apply | 3 |
|  | 3.3. | Demand Paging. | Understand | 1 |
|  | 3.4. | Page Replacement : Page Replacement Algorithm | Apply | 3 |
|  | 3.5. | Virtual Memory. | Understand | 2 |
|  | 3.6. | File Concepts, Access Methods | Understand | 2 |
|  | 3.7. | File Systems Structures | Understand | 2 |
|  | 3.8. | File System Implementation | Understand | 2 |
|  | 3.9. | Free Space Management | Understand | 2 |
| **4** | **Disk Scheduling and Distributed Systems** | |  | **12** |
|  | 4.1. | I/O Hardware | Understand | 2 |
|  | 4.2. | Application of I/O Interface | Understand | 3 |
|  | 4.3. | Disk Structures | Understand | 1 |
|  | 4.4. | Disk Scheduling Algorithms | Apply | 2 |
|  | 4.5. | Disk Management | Understand | 1 |
|  | 4.6. | RAID Structure | Understand | 2 |
|  | 4.7. | Distributed System Structure | Understand | 2 |
|  | 4.8. | Distributed File Systems | Understand | 2 |
|  | 4.9. | Naming and Transparency | Understand | 2 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **#** | **Module** | **Course Outcomes** | | | | |
|  | | **CO1** | **CO2** | **CO3** | **CO4** | **CO5** |
| 1 | Operating System Basic Concepts-  Overview | √ |  |  |  |  |
| 2 | Process Management and Process Synchronization |  | √ |  |  |  |
| 3 | Memory Management |  |  | √ |  |  |
| 4 | File Systems and I/O Management |  |  |  | √ |  |
| 5 | Basic of Shell Programming |  |  |  |  | √ |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 4.10. | Remote File Accesses | Understand | 2 |
|  | 4.11. | Stateful Versus Stateless Service | Understand | 2 |
| **5** | **Shell Programming** | |  | **10** |
|  | 5.1. | Introduction to shell programming | Understand | 2 |
|  | 5.2. | Various types of Shells and their Comparisons | Understand | 2 |
|  | 5.3. | Command Execution | Apply | 3 |
|  | 5.4. | Command Execution Process | Apply | 3 |
|  | 5.5. | Shell programming in different shells | Apply | 2 |
|  | 5.6. | Decision Making Selections | Apply | 2 |
|  | 5.7. | Concept of Function , Parameter passing and argument | Apply | 2 |
|  | 5.8. | Filters. | Apply | 2 |

|  |  |
| --- | --- |
| Course Modules and Course Outcomes Mapping: |  |
| Textbooks/ Resources: | 1. Abraham Silberschatz, Peter B Galvin, and Gerg Gagne – “Operating System Concepts”, Wiley. 2. Behrouz A. Forouzan and Richard F. Gilberg – “UNIX and Shell Programming”. |
| Reference  Books/ Resources | 1. Tanenbaum A - Modern Operating Systems - Pearson Education. 2. Dhamdhere D. M. – “Operating Systems”, Tata McGraw Hill. 3. Sumitabha Das – “UNIX Concepts and Applications”, Tata McGraw Hill. 4. Halder S. and Aravind A. – “Operating Systems”, Pearson. |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **SYLLABUS**  **PROGRAM: MCA SEMESTER: II WEF: 2023-24** | | | | | | |
| **Course Code: MCA 206** | | **Credit: 04** | | **Elective- I(Course: Data Warehousing and Data Mining)** | **L: 03 | T: 01 | P: 00** | |
| Prerequisites: | Nill | | | | | |
| Objectives: | To understand the need for analysis of large datasets, data mining models and methods to  discover interesting patterns from such dataset. To understand the need for analysis of large dataset, multidimensional data modelling, OLAP and various data mining techniques. | | | | | |
| Course Outcome: | Upon successfully finishing the course, students will have the capability to: | | | | | |
|  | **No.** | | **Course Outcome** | | | **BT Level** |
| CO1 | | Appreciate the multidisciplinary field of data mining, its need and  the importance. | | | Understand |
| CO2 | | Apply various pre-processing techniques on the data before  mining. | | | Analyze,  Apply |
| CO3 | | Understand, design and create a data warehouse and perform  OLAP operations on it. | | | Apply |
| CO4 | | Appreciate and apply the concept of association rule mining. | | | Apply |
| CO5 | | Appreciate and apply the concept of classification and clustering. | | | Apply |



Program Outcomes and Course Outcomes Mapping:

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcomes** | **Program Outcomes** | | | | | | | | | | | |
| PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO  12 |
| CO1 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO2 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO3 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO4 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO5 |  |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | | | | | |
| **No.** | **Module Description** | | | **BT Level** | **Hours** |
| **1** | **Introduction to Data Mining** | | |  | 8 |
|  | 1.1 | Introduction to Data Science: Data mining, Machine Learning, Deep Learning, Artificial Intelligence, Data Warehouse, Big Data | | Understand | 2 |
|  | 1.2 | Data Mining, Knowledge Discovery from Data (KDD) Framework | | Understand | 2 |
|  | 1.3 | Types of data for Data Mining | | Understand | 1 |
|  | 1.4 | Data Mining: Confluence of multiple disciplines | | Understand | 1 |
|  | 1.5 | Data Mining Applications | | Understand | 2 |
| **2** | **Data Preprocessing** | | |  | **10** |
|  | 2.1 | | Data types: Nominal attributes, Binary attributes, Ordinal attributes | Understand | 2 |
|  | 2.2 | | Statistics of data: Central tendency, dispersion of data - Range, quartiles, Variance and standard deviation | Understand | 2 |
|  | 2.3 | | Covariance and correlation analysis | Understand | 2 |
|  | 2.4 | | Data quality, Data cleaning: Missing values, Noisy data, Data integration | Understand | 2 |
|  | 2.5 | | Data transformation: Normalization, Discretization | Understand | 2 |
| **3** | **Data warehousing and Online Analytical Processing** | | |  | **10** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 3.1 | Introduction to Data Warehouse | Analyze | 1 |
|  | 3.2 | Data Warehouses Architecture: The three-tier architecture, ETL, Enterprise data warehouse and data mart | Analyze | 2 |
|  | 3.3 | Data cube: a multidimensional data model | Understand | 2 |
|  | 3.4 | Schemas for multidimensional data models: stars, snowflakes, and fact constellations | Analyze | 2 |
|  | 3.5 | Concept hierarchies | Analyze | 1 |
|  | 3.6 | OLAP operations | Analyze | 2 |
| **4** | **Association Rule Mining** | |  | **10** |
|  | 4.1. | Market basket analysis | Apply | 2 |
|  | 4.2. | Frequent itemset | Apply | 2 |
|  | 4.3. | Apriori algorithm: finding frequent itemset | Apply | 2 |
|  | 4.4. | Generating association rules from frequent itemset | Apply | 2 |
|  | 4.5 | From association analysis to correlation analysis |  | 2 |
| **5** | **Classification and Cluster Analysis** | |  | **12** |
|  | 5.1 | Introduction to Classification | Understand | 1 |
|  | 5.2 | Decision tree induction | Understand | 2 |
|  | 5.3 | Attribute selection measures: Information gain, Gain ratio | Understand | 2 |
|  | 5.4 | Naïve Bayesian classification | Understand | 2 |
|  | 5.5 | Cluster Analysis | Understand | 1 |
|  | 5.6 | Partitioning methods | Understand | 2 |
|  | 5.7 | k-Means: a centroid-based technique | Understand | 2 |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Course Modules and Course Outcomes Mapping: |  |  |  |  | | | | |  |
| **#** | **Module** | **Course Outcomes** | | | | |
|  | | **CO1** | **CO2** | **CO3** | **CO4** | **CO5** |
| 1 | Introduction to Data Mining | √ |  |  |  |  |
| 2 | Data Preprocessing |  | √ |  |  |  |
| 3 | Data warehousing and Online  Analytical Processing |  |  | √ |  |  |
| 4 | Association Rule Mining |  |  |  | √ |  |
| 5 | Classification and Cluster Analysis |  |  |  |  | √ |
| Text Books/ Resources: | 1. Han, J. and Kamber, M. - Data Mining: Concepts & Techniques, 3rd Edition - Morgan Kaufmann Publishers: TB#1 2. Ian H. Witten, Eibe Frank, Mark A. Hall, Data Mining: Practical Machine Learning Tools and Techniques, Morgan Kaufmann Publications | | | | | | | | |
| Reference Books/  Resource | 1. Mohammed J. Zaki Wagner Meira Jr - Data Mining and Machine Learning: Fundamental Concepts and Algorithms 2. Pujari, A. - Data Mining techniques - Universities Press 3. Pudi, V. and Radhakrishnan, P. - Data Mining - Oxford University Press 4.Larose, D. - Data Mining Methods & Models - Wiley-India   5.Berry, M. and Linoff, G. - Data Mining Techniques - Wiley-India | | | | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **SYLLABUS**  **PROGRAM: MCA SEMESTER: II WEF: 2023-24** | | | | | | |
| **Course Code: MCA 208** | | **Credit: 04** | | **Course: Software Testing** | **L: 03 | T: 01 | P: 00** | |
| Prerequisites: | The prerequisites for a software testing syllabus include an understanding of software development life cycles, testing methodologies, types of testing, test case design, test planning, and the use of testing tools and techniques.s | | | | | |
| Objectives: | The objective of software testing is to ensure that a software application is free of defects, meets specified requirements, and performs reliably and securely under expected conditions. | | | | | |
| Course Outcome: | Upon successfully finishing the course, students will have the capability to: | | | | | |
|  | **No.** | | **Course Outcome** | | | **BT Level** |
| CO1 | | Understand the fundamentals of software testing and its importance in the software development lifecycle | | | Understand |
| CO2 | | Explain the various stages of the software testing process and its lifecycle. | | | Analyze, Apply |
| CO3 | | Demonstrate the application of different test design techniques to create effective test cases. | | | Apply |
| CO4 | | Identify and differentiate between various types of software testing methods and their applications. | | | Apply |
| CO5 | | Gain proficiency in using automated testing tools to improve testing efficiency and accuracy | | | Apply |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **#** | **Module** | **Course Outcomes** | | | | |
|  | | **CO1** | **CO2** | **CO3** | **CO4** | **CO5** |
| 1 | Introduction to Software Testing | √ |  |  |  |  |
| 2 | Testing Process and Life Cycle |  | √ |  |  |  |
| 3 | Test Design Techniques |  |  | √ |  |  |
| 4 | Types of Testing |  |  |  | √ |  |
| 5 | Automated Testing |  |  |  |  | √ |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Program Outcomes and Course Outcomes Mapping: |  | | | | | | | | | | | | |
| **Course Outcomes** | **Program Outcomes** | | | | | | | | | | | |
| PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO2 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO3 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO4 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO5 |  |  |  |  |  |  |  |  |  |  |  |  |
|  | | | | | | | | | | | | |
| Course Modules and Course Outcomes Mapping: |  | | | | | | | | | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Syllabus** | | | | |
| **No.** | **Module Description** | | **BT Level** | **Hours** |
| **1** | **Introduction to Software Testing** | |  | **12** |
|  | 1.1 | Definition of Software Testing: Importance and objectives. | Understand | 2 |
|  | 1.2 | Software Development Life Cycle (SDLC)**:** Role of testing in different SDLC models (Waterfall, Agile, V-Model, Spiral). | Understand | 3 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 1.3 | Levels of Testing**:** Unit testing, Integration testing, System testing, and Acceptance testing. | Understand | 3 |
|  | 1.4 | Types of Testing**:** Manual vs Automated Testing. | Understand | 2 |
|  | 1.5 | Error, Fault, and Failure: Understanding the differences between them | Understand | 2 |
| **2** | **Testing Process and Life Cycle** | |  | **14** |
|  | 2.1 | Testing Process: Requirement analysis, Test planning, Test design, Test execution, Defect reporting, and Closure. | Understand | 3 |
|  | 2.2 | Test Levels**:** Unit testing, Integration testing, System testing, User acceptance testing (UAT). | Understand | 4 |
|  | 2.3 | Test Documentation**:** Test plan, Test case design, Test scripts, Test reports. | Understand | 3 |
|  | 2.4 | Defect Life Cycle: Steps from defect detection to closure. | Understand | 2 |
|  | 2.5 | Test Case Design**:** Writing effective test cases and using test case design techniques. | Understand | 2 |
| **3** | **Test Design Techniques** | |  | **12** |
|  | 3.1 | Black-box Testing**:** Equivalence partitioning, Boundary value analysis, Decision tables, and State transition testing. | Analyze | 3 |
|  | 3.2 | White-box Testing**:** Code-based testing techniques such as statement coverage, branch coverage, path coverage. | Analyze | 3 |
|  | 3.3 | Experience-based Testing**:** Exploratory testing, Error guessing, and Ad-hoc testing. | Understand | 3 |
|  | 3.4 | Test Case Design Techniques: Writing test cases based on requirements and use cases. | Analyze | 3 |
| **4** | **Types of Testing** | |  | **10** |
|  | 4.1. | **Functional Testing**: Focus on the functionality of the software, ensuring the system meets specified requirements. Smoke Testing, Sanity Testing, Regression Testing, Retesting, UAT, and Interface Testing. | Apply | 4 |
|  | 4.2. | Non-Functional Testing, Testing non-functional aspects like performance, security, usability, etc,Performance Testing, Load Testing, Stress Testing, Scalability Testing, and Security Testing. | Apply | 4 |
|  | 4.3. | Regression Testing: Ensuring that new code changes do not affect the existing system. | Apply | 2 |
|  | 4.4. | Exploratory Testing**:** Simultaneously learning about the system and testing it. | Apply |  |
| **5** | **Automated Testing** | |  | **12** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 5.1 | Automation Introduction: Importance, benefits, and challenges. | Understand | 3 |
|  | 5.2 | Tools for Automation: Selenium, QTP, JUnit, TestNG, Appium, etc. | Understand | 3 |
|  | 5.3 | Automated Test Script Design**:** How to design maintainable and reusable test scripts. | Understand | 3 |
|  | 5.4 | Continuous Integration/Continuous Testing: Integration of automated testing in CI/CD pipelines. | Understand | 3 |

|  |  |
| --- | --- |
| Text Books/ Resources: | 1. "Software Testing: Principles and Practices" by M.N. Rao 2. "Foundations of Software Testing: ISTQB Certification" by Aditya P. Mathur 3. "Software Testing: A Craftsman's Approach" by Paul C. Jorgensen |
| Reference Books/  Resource | 1. "How to Break Software: A Practical Guide to Testing" by James Whittaker 2. "The Complete Software Developer's Guide to Testing" by Adam Goucher |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **SYLLABUS**  **PROGRAM: MCA SEMESTER: II WEF: 2023-24** | | | | | | |
| **Course Code: MCA 210** | | **Credit: 04** | | **Elective\_II Course: Advanced Networking Concepts** | **L: 03 | T: 01 | P: 00** | |
| Prerequisites: | Nill | | | | | |
| Objectives: | To understand the concepts and use of data communication and Networking, Goals and applications of networks, OSI and TCP/IP model, function of different layers of model, protocols, basic concept of network security etc. | | | | | |
| Course Outcome: | Upon successfully finishing the course, students will have the capability to: | | | | | |
|  | **No.** | | **Course Outcome** | | | **BT Level** |
| CO1 | | Students are able to understand the basic concepts, services and role of each layer of OSI model and TCP/IP, transmission media ad multiplexing techniques. | | | Understand |
| CO2 | | Students get proficiency to the functions of Data link Layer i.e. flow control, error control, error detection, error correction, noiseless and noisy channel etc. | | | Analyze, Apply |
| CO3 | | Students get proficiency to the functions of Network Layer i.e. IPv4 and IPv6 addressing, subnetting etc. | | | Apply |
| CO4 | | Students get proficiency to the functions of transport layer and Application Layer. | | | Apply |
| CO5 | | Students are able to understand the basic concepts of Network Security and Cryptography. | | | Apply |
| Program Outcome: | Upon successfully finishing the program, students shall be able to: | | | | | |



Program Outcomes and Course Outcomes Mapping:

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcomes** | **Program Outcomes** | | | | | | | | | | | |
| PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO2 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO3 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO4 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO5 |  |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Module Description** | | **BT Level** | **Hours** |
| **1** | **Introduction to Networking and Physical layer** | |  | **10** |
|  | 1.1 | Data Communication: Components like Sender, receiver, protocols etc. | Understand | 1 |
|  | 1.2 | Network Models: OSI and TCP/IP , Protocol | Analyze | 3 |
|  | 1.3 | Addressing : Physical, Logical, Port | Analyze | 2 |
|  | 1.4 | Transmission Media: Wired media, Wireless media | Understand | 2 |
|  | 1.5 | Multiplexing Techniques; Frequency Division, Time Division, Synchronous time- division, Statistical Time Division | Analyze | 1 |
| **2** | **Data link layer** | |  | **15** |
|  | 2.1 | Type of error, redundancy, detection and correction | Understand | 3 |
|  | 2.2 | Block Coding: Error Detection, Error correction , hamming distance | Analyze | 3 |
|  | 2.3 | Cyclic code redundancy and checksum | Analyze | 3 |
|  | 2.4 | Flow control and error control | Understand | 3 |
|  | 2.5 | Noiseless and Noisy channels | Understand | 3 |
| **3** | **Network layer** | |  | **13** |
|  | 3.1 | Logical addressing : IPv4 addressing, IPv6 Addressing | Analyze | 3 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 3.2 | IPv4: Datagram, Fragments, Checksum | Analyze | 4 |
|  | 3.3 | IPv6: Advantages, Packet Format, Extension | Analyze | 3 |
|  | 3.4 | Address Mapping, ICMP,IGMP | Understand | 3 |
| **4** | **Transport layer and Application layer** | |  | **10** |
|  | 4.1. | Process To Process Delivery | Understand | 2 |
|  | 4.2. | TCP and UDP Protocol | Understand | 2 |
|  | 4.3. | Name Space, Domain Name Space | Understand | 2 |
|  | 4.4. | DNS, Resolution | Understand | 2 |
|  | 4.5 | Application layer protocols: SMTP, FTP, POP, IMAP | Understand | 2 |
| **5** | **Network Security and Cryptography** | |  | **12** |
|  | 5.1 | Introduction to security services | Understand | 3 |
|  | 5.2 | Digital signature | Understand | 3 |
|  | 5.3 | Introduction to cryptography: Types | Understand | 3 |
|  | 5.4 | IP Security | Understand | 3 |

|  |  |
| --- | --- |
| Text Books/  Resources: | 1. Forouzan B., “Data Communication and Networking”, 4rd Edition, McGraw-Hill :**TB#1** 2. Andrew S. Tanenbaum, “Computer Networks” , 5th Edition, Person Publication : **TB#2** |
| Reference Books/  Resource | 1. James Kurose, Keith Ross, “Computer Networking: A Top-Down Approach”, 7th Edition, Pearson Publication. 2. Russ White, Ethan Banks, “Computer Networking Problems and Solutions ”, 1st Edition, Cisco Press. 3. https://[www.javatpoint.com/computer-network-tutorial](http://www.javatpoint.com/computer-network-tutorial) 4. https://[www.geeksforgeeks.org/computer-network-tutorials/](http://www.geeksforgeeks.org/computer-network-tutorials/) |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **SYLLABUS**  **PROGRAM: MCA SEMESTER: II WEF: 2023-24** | | | | | | |
| **Course Code: MCA211** | | **Credit:04** | | **Course: Elective-II(Digital Image Processing)** | **L: 03 | T: 01 | P: 00** | |
| Prerequisi tes: | The prerequisites for Digital Image Processing include a solid understanding of mathematics (especially linear algebra and calculus), discrete mathematics, signal processing, computer programming, and basic concepts of image representation. | | | | | |
| Objectives  : | The objective of Digital Image Processing is to enhance, analyze, and manipulate digital images to extract useful information, improve quality, or enable automated interpretation for various applications. | | | | | |
| Course Outcome: | Upon successfully finishing the course, students will have the capability to: | | | | | |
|  | **No.** | | **Course Outcome** | | | **BT Level** |
| CO1 | | Understand the basic principles and techniques of digital image processing and its applications. | | | Understand |
| CO2 | | Apply various image enhancement techniques to improve the visual quality of images. | | | Analyze, Apply |
| CO3 | | Demonstrate the ability to restore degraded images using various restoration methods. | | | Apply |
| CO4 | | Implement thresholding techniques for image segmentation and object extraction. | | | Apply |
| CO5 | | Utilize morphological operations to analyze and process image structures. | | | Apply |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **#** | **Module** | **Course Outcomes** | | | | |
|  | | **CO1** | **CO2** | **CO3** | **CO4** | **CO5** |
| 1 | Introduction to Digital Image  Processing | √ |  |  |  |  |
| 2 | Image Enhancement |  | √ |  |  |  |
| 3 | Image Restoration |  |  | √ |  |  |
| 4 | Thresholding Techniques |  |  |  | √ |  |
| 5 | Morphological Image Processing |  |  |  |  | √ |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Program Outcomes and Course Outcomes Mapping: |  | | | | | | | | | | | | |
| **Course Outcomes** | **Program Outcomes** | | | | | | | | | | | |
| PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO2 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO3 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO4 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO5 |  |  |  |  |  |  |  |  |  |  |  |  |
|  | | | | | | | | | | | | |
| Course Modules and Course Outcomes Mapping: |  | | | | | | | | | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Syllabus** | | | | |
| **No.** | **Module Description** | | **BT Level** | **Hours** |
| **1** | **Introduction to Digital Image Processing** | |  | **15** |
|  | 1.1 | Overview of Digital Image Processing | Understand | 2 |
|  | 1.2 | What is Image Processing? Applications and Examples, Image  Representation: Pixels, Resolution, and Color Models | Understand | 3 |
|  | 1.3 | Types of Images (Gray-scale, RGB, Binary, and Multispectral) | Understand | 3 |
|  | 1.4 | Basic Image Operations,Image Sampling and Quantization,Image Representation in Memory,Image File Formats (BMP, JPEG, PNG, TIFF, etc.) | Understand | 4 |
|  | 1.5 | Introduction to Image Processing Tools,MATLAB/Octave, Python (OpenCV, NumPy), ImageJ | Understand | 3 |
| **2** | **Image Enhancement** | |  | **15** |
|  | 2.1 | Point Processing Operations, Contrast Adjustment (Brightness, Histogram  Equalization),Thresholding and Image Binarization | Understand | 5 |
|  | 2.2 | Spatial Domain Filtering,Smoothing Filters (Mean, Gaussian  Filters),Sharpening Filters (Laplacian, Sobel, Prewitt)Edge | Understand | 5 |
|  | 2.3 | Frequency Domain Processing,Fourier Transform and Image Filtering,Frequency Filtering (Low-pass, High-pass filters),Image  Compression (Lossless, Lossy Compression Techniques) | Understand | 5 |
| **3** | **Image Restoration** | |  | **12** |
|  | 3.1 | Noise Models and Types | Analyze | 3 |
|  | 3.2 | Types of Noise (Gaussian, Salt and Pepper, Speckle), Noise Removal  Techniques (Median Filtering, Wiener Filtering) | Analyze | 4 |
|  | 3.3 | Image Deconvolution, Degradation Models (Blur, Motion Blur),Inverse Filtering and Wiener Deconvolution .Blind Deconvolution and  Regularization Techniques | Understand | 5 |
| **4** | **Thresholding Techniques** | |  | **13** |
|  | 4.1. | Thresholding Techniques , Global Thresholding (Otsu’s Method),Adaptive Thresholding. | Apply | 4 |
|  | 4.2. | Edge-based Segmentation, Canny Edge Detection, Region Growing and Watershed Segmentation | Apply | 4 |
|  | 4.3. | Clustering-based Segmentation, k-Means Clustering, Mean-Shift  Segmentation, Region-based Segmentation, Active Contours (Snakes)Graph- based Segmentation Techniques | Apply | 5 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **5** | **Morphological Image Processing** | |  | **5** |
|  | 5.1 | Basic Morphological Operations, Dilation, Erosion, Opening, and Closing Hit- or-Miss Transform | Understand | 1 |
|  | 5.2 | Advanced Morphological Techniques | Understand | 2 |
|  | 5.3 | Applications of Morphology, Shape Analysis and Object Recognition, Noise Removal and Object Detection | Understand | 2 |

|  |  |
| --- | --- |
| Text Books/ Resources: | 1."Digital Image Processing" by Rafael C. Gonzalez and Richard E. Woods 2."Fundamentals of Digital Image Processing" by Anil K. Jain  3."Digital Image Processing: A Practical Introduction Using Java" by Wilhelm Burger and Mark J. Burge |
| Reference Books/  Resource | 1. "The Image Processing Handbook" by John C. Russ 2. "Handbook of Digital Signal Processing: Engineering Applications" by Douglas S. Eide |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **SYLLABUS**  **PROGRAM: MCA SEMESTER: III WEF: 2024-25** | | | | | |
| **Course Code: 301** | | **Credit: 04** | | **Course: Software Project Management** | **L: 03 | T: 01 |**  **P: 00** |
| Prerequisites: | Nil | | | | |
| Objectives: | Equip them with essential skills in planning, executing, and managing software projects. It  focuses on understanding project management fundamentals, developing planning skills, and using tools for effective project control and risk management. | | | | |
| Course  Outcome: | Upon successfully finishing the course, students will have the capability to: | | | | |
|  | **No.** | | **Course Outcome** | | **BT Level** |
| CO1 | | Students will master project planning, scheduling, and estimation techniques, including using various project management tools and addressing common planning and estimation issues. | | Understanding  ,  Applying,  Analyzing, Evaluating |
| CO2 | | Students will develop skills in eliciting, specifying, and managing software requirements, using use cases, and handling changes and issues effectively. | | Applying, Analyzing, Evaluating |
| CO3 | | Students will understand the SDLC phases and various models such as Waterfall, Agile, and Spiral, applying them to manage and execute software projects effectively. | | Understanding  ,  Applying, Analyzing,  Evaluating |
| CO4 | | Students will gain proficiency in design reviews, refactoring, unit testing, and automation, while also learning to diagnose design and programming problems and manage software  testing processes. | | Understanding  ,  Applying |
| CO5 | | Students will learn to create and execute test plans, track defects, and apply process improvement strategies, while | | Understanding  ,  Applying, |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | understanding the challenges of change management and  enhancing software testing practices. | | | | | | | | | | | Analyzing,  Evaluating | | |
| Program Outcomes and Course Outcomes Mapping: |  | | | | | | | | | | | | | | |
| **Course Outcomes** | | **Program Outcomes** | | | | | | | | | | | | |
| PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | | PO11 | PO12 |
| CO1 | |  |  |  |  |  |  |  |  |  |  | |  |  |
| CO2 | |  |  |  |  |  |  |  |  |  |  | |  |  |
| CO3 | |  |  |  |  |  |  |  |  |  |  | |  |  |
| CO4 | |  |  |  |  |  |  |  |  |  |  | |  |  |
| CO5 | |  |  |  |  |  |  |  |  |  |  | |  |  |
|  | | | | | | | | | | | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Module Description** | | **BT Level** | **Hours** |
| **1** | **Fundamentals of Software Project Planning and Estimation** | | **Understanding, Applying, Analyzing,**  **Evaluating** | **11** |
|  | 1.1 | Introduction | Understanding | 1 |
|  | 1.2 | Tools and Techniques | Applying | 1 |
|  | 1.3 | Software Project Planning | Analyzing | 1 |
|  | 1.4 | Understand the Project Need | Understanding | 1 |
|  | 1.5 | Create the Project Plan | Applying | 1 |
|  | 1.6 | Diagnosing Project Planning Problems | Evaluating | 1 |
|  | 1.7 | Project Estimation | Analyzing | 1 |
|  | 1.8 | Element of Successful Estimate | Analyzing | 1 |
|  | 1.9 | Wideband Delphi Estimation | Applying | 1 |
|  | 1.10 | Other Estimation Techniques | Applying | 1 |
|  | 1.11 | Diagnosing Estimation Problems | Evaluating | 1 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **2** | **Advanced Project Scheduling and Quality Assurance**  **Techniques** | | **Applying, Analyzing, Evaluating** | **10** |
|  | 2.1 | Project Schedule | Applying | 1 |
|  | 2.2 | Building the Project Schedules | Applying | 1 |
|  | 2.3 | Managing Multiple Projects | Analyzing | 1 |
|  | 2.4 | Use the Schedule to Manage Commitments | Analyzing | 1 |
|  | 2.5 | Diagnosing Scheduling Problems | Evaluating | 1 |
|  | 2.6 | Review, Ispections | Applying | 1 |
|  | 2.7 | Deskchecks, Walkthrough | Applying | 1 |
|  | 2.8 | Code Reviews, Pair Programming | Applying | 1 |
|  | 2.9 | Use Inspections to Manage Commitments | Analyzing | 1 |
|  | 2.10 | Diagnosing Review Problems | Evaluating | 1 |
| **3** | **Comprehensive Software Requirements Management** | | **Understanding,**  **Applying, Analyzing** | **7** |
|  | 3.1 | Software Requirements | Understanding | 1 |
|  | 3.2 | Requirement Elicitation | Applying | 1 |
|  | 3.3 | Use Case | Applying | 1 |
|  | 3.4 | Software Requirements Specification | Analyzing | 1 |
|  | 3.5 | Change Control | Analyzing | 1 |
|  | 3.6 | Introduce Software Requirements Carefully | Applying | 1 |
|  | 3.7 | Diagnosing Software Requirement Problems | Evaluating | 1 |
| **4** | **Overview of Software Development Life Cycle (SDLC) Models** | | **Understanding,**  **Applying** | **6** |
|  | 4.1 | Software Development File Cycle: Introduction, Software Process | Understanding | 1 |
|  | 4.2 | SDLC Life Cycle, Need of SDLC, | Understanding | 1 |
|  | 4.3 | SDLC Models | Applying | 1 |
|  | 4.4 | Waterfall Model, RAD Model, Spiral Model | Applying | 1 |
|  | 4.5 | V-Model, Incremental Model, Agile Model | Applying | 1 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 4.6 | Iterative Model, Big-Bang Model, Prototype Model | Applying | 1 |
| **5** | **Advanced Software Design, Testing, and Process Improvement** | | **Understanding,**  **Applying, Analyzing, Evaluating, Creating** | **13** |
|  | 5.1 | Design and Programming | Analyzing | 1 |
|  | 5.2 | Review the Design, Vision Control with Subversion | Applying | 1 |
|  | 5.3 | Refactoring, Unit Testing, Use Automation, Be Careful with Existing Projects | Evaluating | 1 |
|  | 5.4 | Diagnosing Design and Programming Problems | Evaluating | 1 |
|  | 5.5 | Software Tesing: Test Plan and Test Case, Test Execution | Applying | 1 |
|  | 5.6 | Defet Tracking and Triage | Analyzing | 1 |
|  | 5.7 | Test Environment and Performance Testing | Analyzing | 1 |
|  | 5.8 | Smoke Test, Test Automation, Postmortem Reports | Applying | 1 |
|  | 5.9 | Using Software Testing Effectively | Analyzing | 1 |
|  | 5.10 | Diagnosing Software Testing Problem | Evaluating | 1 |
|  | 5.11 | Understanding Change: Why Change Fails, How make  Change Succeed | Understanding | 1 |
|  | 5.12 | Process Improvement: Life without a Software Process,  Software Process Improvement | Analyzing | 1 |
|  | 5.13 | Moving Forward | Creating | 1 |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Course |  |  |  |  | | | | |  |
| **#** | **Module** | **CourseOutcomes** | | | | |
| Modules and |
| Course |
|  | | **CO1** | **CO2** | **CO3** | **CO4** | **CO5** |
| Outcomes Mapping: |
| 1 | Fundamentals of Software Project Planning and Estimation | √ |  |  |  |  |
|  | 2 | Advanced Project Scheduling and |  |  |  |  |  |
|  |  | Quality Assurance Techniques | √ |
|  | 3 | Comprehensive Software Requirements Management |  |  | √ |  |  |
|  | 4 | Advanced Software Design, Testing, |  |  |  | √ |  |
|  |  | and Process Improvement |  |
|  | 5 | Advanced Software Design, Testing, and Process Improvement |  |  |  |  | √ |
| Textbooks/ Resources: | 1. Andrew Stellman and Jennifer Greene, “Applied Software Project Management”, O’REILLY: #TB1 2. https://[www.javatpoint.com/software-engineering-software-development-life-cycle](http://www.javatpoint.com/software-engineering-software-development-life-cycle) | | | | | | | | |
| Reference  Books  /Resources | 1. Jack T. Marchewka, “Information Technology Project Management”, WILEY 2. Bob Hughes and Mike Cotterell, “Software Project Management”, McGraw Hill | | | | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **SYLLABUS**  **PROGRAM: MCA SEMESTER: III WEF: 2024-25** | | | | | |
| **Course Code: MCA 302** | | **Credit: 04** | **Elective –I Course : MACHINE LEARNING** | **L: 03 | T: 01 | P: 00** | |
| Prerequisites: | Nil | | | | |
| Objectives: | The course objectives of a machine learning (ML) are to Gain a solid understanding of fundamental concepts in machine learning, including supervised learning, unsupervised learning, and reinforcement learning. Learn various machine learning algorithms and models, such as linear regression, decision trees, support vector machines, neural networks, clustering algorithms, and ensemble methods. | | | | |
| Course Outcome: | Upon successfully finishing the course, students will have the capability to: | | | | |
|  | **No.** | **Course Outcome** | | | **BT Level** |
| CO1 | Understand the definitions and fundamental terms used in machine learning, as well as the differences between supervised, unsupervised, and reinforcement learning. | | | Understan d |
| CO2 | Gain proficiency in regression and implement various types of regression techniques. | | | Analysis |
| CO3 | Understand Classification and implement various classification techniques like SVM, Naïve Bayes, K-NN etc. | | | Analysis |
| CO4 | Understand unsupervised learning and implement various algorithm. | | | Understan d |
| CO5 | Apply machine learning techniques (NLP) to perform tasks such as text analysis and image recognition. | | | Apply |



Program Outcomes and Course Outcomes Mapping:

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcomes** | **Program Outcomes** | | | | | | | | | | | |
| PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO2 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO3 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO4 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO5 |  |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Curriculum** | | | | |
| **No.** | **Module Description** | | **BT Level** | **Hours** |
| 1 | **Introduction to Machine Learning Techniques** | |  | **10** |
|  | 1.1 | Introduction and Future Scope of Machine Learning | Understand | 1 |
|  | 1.2 | Techniques of Machine Learning | Understand | 2 |
|  | 1.3 | Application of Machine Learning | Understand,  Analyse | 2 |
|  | 1.4 | Types of data in Machine Learning | Understand | 2 |
|  | 1.5 | Data Preprocessing Techniques | Understand,  Analyse | 1 |
|  | 1.6 | Continues and discrete data Dimensionality Reduction | Understand | 2 |
| **2** | **Regression** | |  | **10** |
|  | 2.1 | Introduction to Regression | Understand | 1 |
|  | 2.2 | Types of Regression model | Understand | 1 |
|  | 2.3 | Simple Linear Regression | Understand | 1 |
|  | 2.4 | Multiple Linear Regression | Understand | 1 |
|  | 2.5 | Polynomial Regression | Understand | 1 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 2.6  2.7 | Decision Tree Regression Random Forest Regression | Understand Understand | 1  1 |
|  | 2.8 | Model Building using Least squares | Understand,  Analyse | 2 |
|  | 2.9 | Model Accuracy & Selection | Understand,  Analyse | 1 |
| 3 | **Classification** | |  | **11** |
|  | 3.1 | Introduction to Classification | Understand | 1 |
|  | 3.2 | Logistic Regression | Understand | 2 |
|  | 3.3 | K-Nearest Neighbors (K-NN) | Understand | 2 |
|  | 3.4 | Support Vector Machine (SVM) | Understand | 2 |
|  | 3.5 | Naive Bayes | Understand | 2 |
|  | 3.6 | Ensemble Methods | Understand,  Analyse | 2 |
| 4 | **Unsupervised Learning & Association Rule Mining** | |  | **18** |
|  | 4.1 | Introduction to Unsupervised learning | Understand | 2 |
|  | 4.2 | K-Means Algorithm | Understand | 2 |
|  | 4.3 | K-Means – Variations | Understand,  Analyse | 2 |
|  | 4.4 | Detecting Outliers | Understand,  Analyse | 2 |
|  | 4.5 | Hierarchical Clustering | Understand | 2 |
|  | 4.6 | Association Rule Mining | Understand,  Analyse | 2 |
|  | 4.7 | Apriori Algorithm | Understand,  Analyse | 2 |
|  | 4.8 | Time series Prediction | Understand,  Analyse | 2 |
|  | 4.9 | Markov Process | Understand | 1 |
|  | 4.10 | Hidden Markov Model | Understand,  Analyse | 1 |
| **5** | **Natural Language Processing** | |  | **11** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 5.1 | Introduction to NLP | understand | 2 |
|  | 5.2 | Application of NLP | Understand,  Analyse | 2 |
|  | 5.3 | Classical Vs Deep Learning Models | Understand,  Analyse | 2 |
|  | 5.4 | Bag of words model | Understand,  Apply | 3 |
|  | 5.5 | Tools and techniques for NLP | Understand | 2 |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Course |  |  |  |  | | | | |
| **#** |  |  | | | | |
| Modules and | **Module** | **CourseOutcomes** | | | | |
| Course |  |  | | | | |
|  | | **CO1** | **CO2** | **CO3** | **CO4** | **CO5** |
| Outcomes Mapping: |
| 1 | Introduction to Machine Learning  Techniques |  |  |  |  |  |
|  | 2 | Regression |  |  |  |  |  |
|  | 3 | Classification |  |  |  |  |  |
|  | 4 | Unsupervised Learning & Association Rule Mining |  |  |  |  |  |
|  |  |  |
|  | 5 | Natural Language Processing |  |  |  |  |  |
|  | | | | | | | |
| Text Books/ Resources: | 1. Saikat Dutt, Subramanian Chandramouli, Amit Kumar Das, “Machine Learning”, Pearson 2. "Machine Learning: A Probabilistic Perspective" by Kevin P. Murphy 3. "Deep Learning" by Ian Goodfellow, Yoshua Bengio, and Aaron Courville | | | | | | | |
| Reference  Books/ Resources | 1. Aurelien Geron ,“Hands–On Machine Learning with Scikit–Learn and TensorFlow”(3rd Edition”) Paperback, 2017. 2. Sebastian Raschka ,Vahid Mir Jalili, “Python Machine Learning”, Third Edition Packt   . | | | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **SYLLABUS**  **PROGRAM: MCA SEMESTER: III WEF: 2024-25** | | | | | | |
| **Course Code:**  **MCA 304** | | **Credit: 04** | | **Elective –II Course: Artificial Intelligence** | **L: 03 | T: 01 | P: 00** | |
| Prerequisites: | Basic Mathematics and Logical Thinking | | | | | |
| Objectives: | Understanding the fundamentals of digital electronics is crucial for anyone working in fields such as computer science, electrical engineering, or information technology. | | | | | |
| Course  Outcome: | Upon successfully finishing the course, students will have the capability to: | | | | | |
|  | **No.** | | **Course Outcome** | | | **BT Level** |
| CO1 | | Introduce students to the concepts and principles of Artificial Intelligence (AI). | | | Understand, Apply |
| CO2 | | Explain the algorithms in the various subfields of AI. | | | Understand,  Apply, |
| CO3 | | Understand and apply basic problem-solving strategies such as divide and conquer, greedy algorithms, and dynamic programming. | | | Understand,  Apply, Analyze |
| CO4 | | Implement various search and traversal algorithms such as breadth- first search, depth first search, and binary search. | | | Understand, Apply,  Analyze |
| CO5 | | Familiarize students with the history and evolution of AI and its impact on society | | | Understand, Apply,  Analyze |



Program Outcomes and Course Outcomes Mapping:

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcomes** | **Program Outcomes** | | | | | | | | | | | |
| PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO2 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO3 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO4 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO5 |  |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Module Description** | | **BT Level** | **Hours** |
| **1** | **Introduction to Artificial Intelligence.** | |  | 8 |
|  | 1.1 | Artificial Intelligence | Understand | 1 |
|  | 1.2 | The Foundations of Artificial Intelligence. | Understand,  Apply | 1 |
|  | 1.3 | The History of Artificial Intelligence, AI Techniques, advantages, and limitations of AI, Impact and Examples of AI, Application domains of  AI. | Understand, Apply | 1 |
|  | 1.4 | The State of the Art. | Understand, Apply | 1 |
|  | 1.5 | Intelligent Agents: Agents and Environments. | Understand,  Apply | 2 |
|  | 1.6 | Good Behavior: The Concept of Rationality. | Understand,  Apply | 1 |
|  | 1.7 | The Nature of Environments. | Understand,  Apply | 1 |
|  | 1.8 | The Structure of Agents. | Understand,  Apply | 1 |
| **2** | **Introduction to Algorithms in Artificial Intelligence.** | |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 2.1 | Introduction to Algorithms in Artificial Intelligence: Definition | Understand | 1 |
|  | 2.2 | Significance of algorithms in artificial intelligence. | Understand, Apply | 1 |
|  | 2.3 | Basic components of an algorithm: input, output, | Understand | 1 |
|  | 2.4 | Control structures. | Understand,  Apply | 1 |
|  | 2.5 | Introduction to problem-solving techniques in Artificial Intelligence | Understand,  Apply | 2 |
|  | 2.6 | Knowledge representation. | Understand,  Apply | 2 |
|  | 2.7 | Reasoning. | Understand, Apply | 2 |
| **3** | **Search Algorithms** | |  |  |
|  | 3.1 | Introduction to search algorithms | Understand,  Analyze | 1 |
|  | 3.2 | Depth-first search. | Understand, Analyze | 1 |
|  | 3.3 | Breadth-first search | Understand,  Analyze | 1 |
|  | 3.4 | Heuristic search techniques | Understand,  Analyze | 2 |
|  | 3.5 | A\* search | Understand,  Analyze | 2 |
|  | 3.6 | Hill climbing, Min-Max Search, Alpha-Beta. | Understand,  Analyze | 1 |
| **4** | **Introduction to neural networks and Expert System** | |  |  |
|  | 4.1. | Models of neuron McCulloch – Pitts model. | Understand,  Analyze | 2 |
|  | 4.2. | Perceptron, Adaline model, Basic learning laws. | Understand,  Apply | 2 |
|  | 4.3. | Topology of neural network architecture, Multilayer Neural Networks,  Learning Methods. | Understand,  Apply | 3 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 4.4 | The human element in expert systems. | Understand,  Apply | 3 |
| **5** | **Introduction of Fuzzy logic and Neuro Fuzzy Systems and Case Study with Artificial**  **Intelligence** | |  |  |
|  | 5.1 | Fuzzy sets, Fuzzy model, Fuzzy rule generation Fuzzy inference system. | Understand,  Apply | 2 |
|  | 5.2 | Integrating Artificial Intelligence in Yoga Applications. | Understand,  Analyze | 2 |
|  | 5.3 | Personalized yoga routines based on user preferences and goals. | Understand,  Analyze | 2 |
|  | 5.4 | AI-powered progress tracking and performance analytics. | Understand,  Analyze | 3 |
|  | 5.5 | Design considerations for AI-enhanced yoga applications. | Understand,  Apply | 3 |

|  |  |
| --- | --- |
| Text Books/ Resources: | 1"Artificial Intelligence" by Stuart Russell and Peter Norvig, Third Edition 2010, Pearson Education, Inc.   1. "Artificial Intelligence: Foundations of Computational Agents" by David L. Poole and Alan K. Mackworth. 2. "Introduction to Artificial Intelligence and Expert Systems" by Dan W. Patterson, Publisher.   Pearson Education India, 2015, Pages 466 |
| Reference Books/ Resource | 1."Artificial Intelligence" by Rich, E., Knight, K., & Nair, S. (2009), Tata McGraw Hill. 2" AI Algorithms Lab: Hands-On Exercises in Artificial Intelligence" by John Smith (Year: 2023). |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **SYLLABUS**  **PROGRAM: MCA SEMESTER: III WEF: 2023-24** | | | | | | |
| **Course Code: MCA306** | | | **Credit: 3** | **Course: Elective-I (Responsive and User Experience(UX)Web Design** | **L: 03 | T: 01 | P: 00** | |
| Prerequisites: | The prerequisites for responsive and user experience (UX) web design include proficiency in HTML, CSS, and JavaScript, understanding mobile-first and flexible design principles, conducting user research, ensuring accessibility and usability, and optimizing performance across devices and browsers to create seamless, intuitive, and visually appealing web experiences. | | | | | |
| Objectives: | The objective of Responsive and User Experience (UX) Web Design is to create websites that provide seamless, accessible, and intuitive experiences across all devices, ensuring user satisfaction and engagement. | | | | | |
| Course Outcome: | Upon successfully finishing the course, students will have the capability to: | | | | | |
|  | **No.** | **Course Outcome** | | | | **BT Level** |
| CO1 | Students will gain a foundational understanding of web design concepts and basic principles. | | | | Understand |
| CO2 | Students will be able to create and style basic web pages using HTML and CSS. | | | | Apply |
| CO3 | Students will learn to design websites that adapt to various screen sizes and devices using responsive design techniques. | | | | Apply |
| CO4 | Students will master advanced CSS techniques to enhance the user experience and interface design. | | | | Apply |
| CO5 | Students will apply key UX principles to design intuitive and user-centered websites. | | | | Analyze |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Program Outcomes and Course Outcomes Mapping: | | |  | | | | | | | | | | | | | | | |
| **Course Outcomes** | **Program Outcomes** | | | | | | | | | | | | | | |
| PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | | PO10 | PO11 | | PO12 | |
| CO1 |  |  |  |  |  |  |  |  |  | |  |  | |  | |
| CO2 |  |  |  |  |  |  |  |  |  | |  |  | |  | |
| CO3 |  |  |  |  |  |  |  |  |  | |  |  | |  | |
| CO4 |  |  |  |  |  |  |  |  |  | |  |  | |  | |
| CO5 |  |  |  |  |  |  |  |  |  | |  |  | |  | |
|  | | | | | | | | | | | | | | | |
| **Curriculum** | | | | | | | | | | | | | | | | | |  |
| **No.** | **Module Description** | | | | | | | | | | | | **BT Level** | | | **Hours** | |  |
| 1 | **Introduction to Web Design** | | | | | | | | | | | |  | | | **10** | |  |
|  | 1.1 | Overview of Web Design, Basic concepts of web design and development, Difference between web design, web development, and user experience (UX), Importance of  responsive design in modern websites | | | | | | | | | | | Understand | | | 1 | |  |
|  | 1.2 | Introduction to UX, Defining User Experience in the context of web design, Understanding user needs and user-centered design  (UCD). | | | | | | | | | | | Understand | | | 2 | |  |
|  | 1.3 | Key principles of UX design: usability, accessibility, and user  satisfaction | | | | | | | | | | | Understand,  Analyse | | | 2 | |  |
| **2** | **HTML and CSS Basics** | | | | | | | | | | | |  | | | **10** | |  |
|  | 2.1 | Structure of HTML documents | | | | | | | | | | | Understand | | | 1 | |  |
|  | 2.2 | Key HTML tags and elements (head, body, navigation, sections,  etc.) | | | | | | | | | | | Understand | | | 1 | |  |
|  | 2.3 | Forms and multimedia elements | | | | | | | | | | | Understand | | | 1 | |  |
|  | 2.4 | CSS Fundamentals, Styling HTML elements (fonts, colors, margins,  etc.), CSS box model (content, padding, border, margin) | | | | | | | | | | | Understand | | | 1 | |  |
|  | 2.5 | CSS positioning (static, relative, absolute, fixed) | | | | | | | | | | | Understand | | | 1 | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 3 | **Responsive Web Design (RWD)** | |  | **11** |
|  | 3.1 | Importance of designing for different screen sizes (desktop,  tablet, mobile) | Understand | 1 |
|  | 3.2 | Fluid layouts, flexible grids, and media queries | Understand | 2 |
|  | 3.3  3.4 | **Building a Responsive Web Layout**, Fluid grids vs. fixed-width layouts, Using CSS media queries to adjust layouts for different screen sizes, Mobile-first design approach  Responsive Images, How to make images responsive using  CStechniques (max-width: 100%), Best practices for optimizing images for different devices | Understand  Understand | 2  2 |
| 4 | **Advanced CSS for UX Design** | |  | **18** |
|  | 4.1 | CSS Transitions and Animations | Understand | 2 |
|  | 4.2 | Basic and advanced animations for smooth transitions (hover  effects, buttons, menus) | Understand | 2 |
|  | 4.3 | Performance considerations for animations on the web | Understand,  Analyse | 2 |
|  | 4.4 | CSS Variables and Custom Properties | Understand,  Analyse | 2 |
|  | 4.5 | Using CSS variables for consistent design across the site | Understand | 2 |
|  | 4.6 | Creating reusable design elements | Understand,  Analyse | 2 |
|  | 4.7 | Typography for Web Design, Web-safe fonts and Google Fonts,Line  height, letter spacing, and font-size adjustments for readability | Understand,  Analyse | 2 |
| **5** | **User Experience (UX) Principles** | |  | **11** |
|  | 5.1 | User Research and Personas | understand | 2 |
|  | 5.2 | Conducting user research and creating user personas | Understand,  Analyse | 2 |
|  | 5.3 | Identifying user pain points and goals | Understand,  Analyse | 2 |
|  | 5.4 | Wireframing and Prototyping, Tools for wireframing (e.g., Figma, Adobe XD, Sketch), Creating low-fidelity and high-fidelity  prototypes | Understand, Apply | 3 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 5.5 | Usability Testing, Types of usability testing (A/B testing, usability  studies), Methods to collect feedback and insights from users | Understand | 2 |

|  |  |
| --- | --- |
| Course Modules and Course Outcomes Mapping: |  |
| Text Books/ Resources: | 1. "UX Design: The Ultimate Guide to User Experience Design" by Joshua S. Porter 2. "HTML and CSS: Design and Build Websites" by Jon Duckett 3. "Seductive Interaction Design: Creating Playful, Fun, and Effective User Experiences" by Stephen Anderson. |
| Reference  Books/  Resources | 1. "Lean UX: Designing Great Products with Agile Teams" by Jeff Gothelf and Josh Seiden 2. "Responsive Web Design" by Ethan Marcotte |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **SYLLABUS**  **PROGRAM: MCA SEMESTER: I WEF: 2023-24** | | | | | | |
| **Course Code: MCA308** | | | **Credit: 3** | **Course: Advanced Web Development (Laravel, React,Angular)** | **L: 03 | T: 01 | P: 00** | |
| Prerequisites: | The prerequisites for advanced web development with Laravel, React, and Angular include a solid understanding of HTML, CSS, JavaScript, PHP, RESTful APIs, and version control with Git, along with experience in object-oriented programming and asynchronous programming. | | | | | |
| Objectives: | The objective of advanced web development with Laravel, React, and Angular is to build scalable, high-performance, and dynamic web applications by leveraging modern frameworks and technologies for efficient front-end and back-end  development. | | | | | |
| Course Outcome: | Upon successfully finishing the course, students will have the capability to: | | | | | |
|  | **No.** | **Course Outcome** | | | | **BT Level** |
| CO1 | Students will be able to implement complex Laravel features such as queues, jobs, and advanced routing in real-world applications. | | | | Understand |
| CO2 | Students will gain proficiency in using advanced Eloquent ORM techniques like relationships, scopes, and custom collections for efficient database management. | | | | Apply |
| CO3 | students will be able to design and build robust RESTful APIs using Laravel, including authentication, validation, and error handling. | | | | Apply |
| CO4 | Students will understand and apply Laravel middleware and service providers to manage request flow and inject dependencies in Laravel applications | | | | Apply |
| CO5 | Students will learn to implement event-driven programming in Laravel using events, listeners, and queues to create scalable and maintainable applications. | | | | Analyze |



Program Outcomes and Course Outcomes Mapping:

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcomes** | **Program Outcomes** | | | | | | | | | | | |
| PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO2 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO3 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO4 |  |  |  |  |  |  |  |  |  |  |  |  |
| CO5 |  |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Syllabus** | | | | |
| **No.** | **Module Description** | | **BT Level** | **Hours** |
| **1** | **Advanced Laravel Development** | |  | **10** |
|  | 1.1 | Introduction to Laravel (Recap), Laravel installation, folder structure. | Understand | 5 |
|  | 1.2 | folder structure, and configuration, Artisan CLI and migrations, Routing, controllers, views, and Blade templating, Eloquent ORM and database interactions |  | 5 |
| **2** | **Advanced Eloquent ORM** | |  | **10** |
|  | 2.1 | Relationships: One-to-One, One-to-Many, Many-to-Many, Polymorphic | Analyze | 3 |
|  | 2.2 | Advanced query building (joins, subqueries, raw queries) | Analyze | 4 |
|  | 2.3 | Custom Eloquent collections and scopes | Understand | 3 |
| **3** | **RESTful API Development with Laravel** | |  | **10** |
|  | 3.1. | Building RESTful APIs in Laravel | Apply | 2 |
|  | 3.2. | Authentication with Passport and Sanctum | Apply | 3 |
|  | 3.3. | Validation, authorization, and error handling | Apply | 3 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 3.4. | API versioning and pagination | Apply | 2 |
| **4** | **Laravel Middleware and Service Providers** | |  | **10** |
|  | 4.1. | Writing custom middleware for request filtering and validation | Understand | 5 |
|  | 4.2. | Understanding service providers and dependency injection | Understand | 5 |
| **5** | **Event-Driven Programming** | |  | **05** |
|  | 5.1. | Event handling and broadcasting | Understand | 2 |
|  | 5.2. | Jobs, queues, and scheduling tasks | Understand | 2 |
|  | 5.3. | Security in Laravel | Understand | 1 |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Course Modules and Course Outcomes Mapping: |  |  |  |  | | | | |
| **#** | **Module** | **Course Outcomes** | | | | |
|  | | **CO1** | **CO2** | **CO3** | **CO4** | **CO5** |
| 1 | Advanced Laravel Development |  |  |  |  |  |
| 2 | Advanced Eloquent ORM |  |  |  |  |  |
| 3 | RESTful API Development with Laravel |  |  |  |  |  |
| 4 | Laravel Middleware and Service Providers |  |  |  |  |  |
| 5 | Event-Driven Programming |  |  |  |  |  |
|  | | | | | | | |
| Textbooks/ Resources: | 1. "Laravel: Up and Running: A Framework for Building Modern PHP Apps" by Matt Stauffer 2. "Modern PHP: New Features and Good Practices" by Josh Lockhart 3."Laravel Design Patterns and Best Practices" by Arda Kılıçdağı, Huseyin Uysal   4. "Angular: Up and Running: Learning Angular, Step by Step" by Shyam Seshadri | | | | | | | |
| Reference Books/  Resources | 1. "Web Development with Node and Express: Leveraging the JavaScript Stack" by Ethan Brown 2. Deigning Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems" by Martin Kleppmann | | | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **SYLLABUS**  **PROGRAM: MCA SEMESTER: III WEF: 2024-25** | | | | | | |
| **Course Code: MCA 314** | | **Credit: 02** | | **Course: Internet Of Things** | **L: 0 | T: 01 | P: 00** | |
| Prerequisites: | Basic Internet Of Things Concepts | | | | | |
| Objectives: | Gain a solid foundation in IoT concepts, including the architecture, protocols, and  Technologies involved. Explore the components of IoT systems such as sensors, actuators, connectivity, and data processing. | | | | | |
| Course  Outcome: | Upon successfully finishing the course, students will have the capability to: | | | | | |
|  | **No.** | | **Course Outcome** | | | **BT Level** |
| CO1 | | Student get proficiency to the Internet Of Things its basic concepts like IoT architecture and it’s concepts, communication model and communication protocols LED, buzzer,LCD. | | | Understand |
| CO2 | | Students gain an understanding of IoT and apply the concepts, including sensors, actuators, and communication protocols. | | | Apply |
| CO3 | | Students are capable of designing and developing IoT applications, managing IoT devices and systems. Also performing operations such as data transmission and device control. | | | Apply |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Program | | |  | | | | | | | | | | | | | | |
| Outcomes and | | | **Course** | **Program Outcomes** | | | | | | | | | | | | | |
| Course | | | **Outcomes** |
| PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | | PO10 | PO11 | | PO12 |
| Outcomes | | |  |
| CO1 |  |  |  |  |  |  |  |  |  | |  |  | |  |
| Mapping: | | |
|  | | | CO2 |  |  |  |  |  |  |  |  |  | |  |  | |  |
|  | | | CO3 |  |  |  |  |  |  |  |  |  | |  |  | |  |
|  | | | CO4 |  |  |  |  |  |  |  |  |  | |  |  | |  |
|  | | | CO5 |  |  |  |  |  |  |  |  |  | |  |  | |  |
| **No.** | | **Module Description** | | | | | | | | | | | | **BT Level** | | | **Hours** | |
| **1** | | **Fundamentals of IoT** | | | | | | | | | | | |  | | | **11** | |
|  | | 1.1 | Introduction, Definitions & Characteristics, Challenges of IoT. | | | | | | | | | | | Understand | | | 1 | |
|  | | 1.2 | IoT Architectures, Physical & Logical Design of IoT, Enabling Technologies in IoT. | | | | | | | | | | | Analyze | | | 2 | |
|  | | 1.3 | History of IoT, About Things in IoT. | | | | | | | | | | | Analyze | | | 2 | |
|  | | 1.4 | The Identifiers in IoT, About the Internet in IoT | | | | | | | | | | | Understand | | | 3 | |
|  | | 1.5 | IoT frameworks, IoT and M2M. | | | | | | | | | | | Analyze | | | 3 | |
| **2** | | **Sensors Networks** | | | | | | | | | | | |  | | | **11** | |
|  | | 2.1 | Definition, Types of Sensors, Types of Actuators, Examples and Working, | | | | | | | | | | | Understand | | | 2 | |
|  | | 2.2 | IoT Development Boards: Arduino IDE and Board Types, Raspberri Pi Development Kit | | | | | | | | | | | Analyze | | | 2 | |
|  | | 2.3 | RFID Principles and components, | | | | | | | | | | | Analyze | | | 2 | |
|  | | 2.4 | History and Context | | | | | | | | | | | Understand | | | 2 | |
|  | | 2.5 | The node, Connecting nodes, Networking Nodes, WSN and IoT. | | | | | | | | | | | Understand | | | 3 | |
| **3** | | **Applications of IoT** | | | | | | | | | | | |  | | | **08** | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 3.1 | Home Automation, Smart Cities, Energy, Retail Management, Logistics, Agriculture, Health and Lifestyle, Industrial IoT | Analyze | 3 |
|  | 3.2 | Legal challenges, IoT design Ethics | Analyze | 3 |
|  | 3.3 | IoT in Environmental Protection. | Analyze | 2 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **#** | **Module** | **Course Outcomes** | | | | |
|  | | **CO1** | **CO2** | **CO3** | **CO4** | **CO5** |
| 1 | Introduction to IoT | ✔ |  |  |  |  |
| 2 | IoT Architecture and Design |  | ✔ |  |  |  |
| 3 | IoT Communication and Networking |  |  | ✔ |  |  |
| 4 | Sensors, Actuators, and Devices, |  |  |  | ✔ |  |
| 5 | IoT Data Management and Cloud Computing |  |  |  |  | ✔ |

|  |  |
| --- | --- |
| Text Books/ Resources: | 1. CunoPfister, “getting started with the Internet of Things : Connecting Sensors and MicroController to the Cloud”kindle2011 : TB#1 2. ArsheepBagha ,Vijay Madisetti,”Internet of Things”:A Hands On |
| Reference Books | Iternet of Things: A Hands-On Approach**"** by Arshdeep Bahga and Vijay Madisetti |

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **SYLLABUS**  **PROGRAM: MCA SEMESTER: IV WEF: 2023-24** | | | | | | | | | | | | |
| **Course MCA402** | **Code:** |  | **Credit: 04** | **Course: Organizational behavior and Professional Ethics** | | | | | | **L: 03 | T: 01 | P: 00** | | |
| Prerequisit es: | Nill | | | | | | | | | | | |
| Objectives: | The objective of Organizational Behavior is to understand and improve individual and group dynamics within an organization to enhance performance, while Professional Ethics aims to promote ethical conduct and decision-making in the workplace. | | | | | | | | | | | |
| Course Outcome: | Upon successfully finishing the course, students will have the capability to: | | | | | | | | | | | |
|  | **No.** | **Course Outcome** | | | | | | | | | | **BT Level** |
| CO1 | Understand | | the | key | concepts | and | theories | related | | to | Understand |
|  | organizational behavior and their application in real-world | | | | | | | | | |  |
|  | settings. | | | | | | | | | |  |
| CO2 | Analyze how individual differences influence behavior and | | | | | | | | | | Apply |
|  | performance within an organization. | | | | | | | | | |  |
| CO3 | Evaluate and apply various motivation theories to enhance | | | | | | | | | | Apply |
|  | employee performance and satisfaction. | | | | | | | | | |  |
| CO4 | Examine how group behavior and team dynamics affect decision- | | | | | | | | | | Apply |
|  | making and productivity in organizations | | | | | | | | | |  |
| CO5 | Identify and assess different leadership styles and power | | | | | | | | | | Analyze |
|  | dynamics that influence organizational success. | | | | | | | | | |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Program | |  | | | | | | | | | | | | | | | |
| Outcomes and Course Outcomes Mapping: | | **Course Outcomes** | | **Program Outcomes** | | | | | | | | | | | | | |
| PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | | PO11 | | PO12 |
| CO1 | |  |  |  |  |  |  |  |  |  |  | |  | |  |
| CO2 | |  |  |  |  |  |  |  |  |  |  | |  | |  |
| CO3 | |  |  |  |  |  |  |  |  |  |  | |  | |  |
| CO4 | |  |  |  |  |  |  |  |  |  |  | |  | |  |
| CO5 | |  |  |  |  |  |  |  |  |  |  | |  | |  |
|  | | | | | | | | | | | | | | | |
| **Syllabus** | | | | | | | | | | | | | | | | | |
| **No.** | **Module Description** | | | | | | | | | | | | | **BT Level** | | **Hours** | |
| **1** | **Introduction to Organizational Behavior** | | | | | | | | | | | | |  | | **15** | |
|  | 1.1 | | Definition and Importance of Organizational Behavior (OB) | | | | | | | | | | | Understand | | 3 | |
|  | 1.2 | | Foundations of OB: Psychological, Social, and Cultural Influences | | | | | | | | | | | Understand | | 3 | |
|  | 1.3 | | Key Concepts in OB: Motivation, Leadership, Communication,  Teamwork, etc. | | | | | | | | | | | Understand | | 3 | |
|  | 1.4 | | Levels of Analysis**:** Individual. | | | | | | | | | | | Understand | | 3 | |
|  | 1.5 | | Group, and Organizational | | | | | | | | | | | Understand | | 3 | |
| **2** | **Individual Behavior in Organizations** | | | | | | | | | | | | |  | | **15** | |
|  | 2.1 | | Personality and its Impact on Work Behavior,Types of Personality  Models (Big Five, MBTI) | | | | | | | | | | | Analyze | | 3 | |
|  | 2.2 | | Perception and Decision Making,Process of Perception,Biases in  Decision Making | | | | | | | | | | | Analyze | | 3 | |
|  | 2.3 | | Attitudes**:** Job Satisfaction, Organizational Commitment, and  Engagement | | | | | | | | | | | Understand | | 3 | |
|  | 2.4 | | Emotions in the Workplace | | | | | | | | | | | Analyze | | 3 | |
|  | 2.5 | | Type of emotions | | | | | | | | | | | Analyze | | 3 | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **3** | **Entrepreneurship and Motivation Theories** | |  | **10** |
|  | 3.1. | Classical Motivation Theories, Maslow’s Hierarchy of Needs  Herzberg’s Two-Factor Theory, McGregor’s Theory X and Y | Apply | 2 |
|  | 3.2. | Entrepreneurship Small Business Entrepreneurship, Scalable Start-  up Entrepreneurship, Social Entrepreneurship and Large Company Entrepreneurship. | Apply | 2 |
|  | 3.3. | Goal Setting Theory (Locke & Latham), | Apply | 2 |
|  | 3.4. | Equity Theory (Adams Self-Determination Theory | Apply | 2 |
|  | 3.5 | Motivational Techniques in Organizations |  | 2 |
| **4** | **Group Behavior and Team Dynamics** | |  | **10** |
|  | 4.1. | Types of Groups**:** Formal and Informal Groups | Understand | 2 |
|  | 4.2. | Group Development Stages (Tuckman’s Model) | Understand | 3 |
|  | 4.3. | Teamwork and Team Effectiveness | Understand | 2 |
|  | 4.4. | Conflict in Groups and Conflict Resolution Strategies | Understand | 3 |
| **5** | **Leadership and Power in Organizations** | |  | **10** |
|  | 5.1. | Leadership Theories, Trait Theory, Behavioral Theories,and Contingency Models (e.g., Fiedler’s Contingency Theory), Transformational and Transactional Leadership, Servant Leadership | Understand | 2 |
|  | 5.2. | Power and Politics in Organizations | Understand | 3 |
|  | 5.3. | Sources of Power, Organizational Politics and Influence Tactics | Understand | 2 |
|  | 5.4. | Ethical Leadership | Understand | 3 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **#** | **Module** | **Course Outcomes** | | | | |
|  | | **CO1** | **CO2** | **CO3** | **CO4** | **CO5** |
| 1 | Introduction to Organizational Behavior |  |  |  |  |  |
| 2 | Individual Behavior in Organizations |  |  |  |  |  |
| 3 | Motivation Theories |  |  |  |  |  |
| 4 | Group Behavior and Team Dynamics |  |  |  |  |  |
| 5 | Leadership and Power in Organizations |  |  |  |  |  |

|  |  |
| --- | --- |
| Course Modules and Course Outcomes Mapping: |  |
| Textbooks/ Resources: | 1. **"**Organizational Behavior" by Stephen P. Robbins and Timothy A. Judge 2. "Organizational Behavior: A Practical, Problem-Solving Approach" by Debra L. Nelson and James Campbell Quick 3. "Essentials of Organizational Behavior" by Stephen P. Robbins and Timothy A. Judge |
| Reference Books/  Resources | 1. "The Moral Compass of the Corporation: Business Ethics and the Path to Corporate Social Responsibility" by Denise Rousseau. 2. "Ethics in the Workplace: Tools and Tactics for Organizational Transformation" by Craig   E. Johnson |