



**Vivekanand Education Society's
College of Arts, Science and Commerce (Autonomous)
Sindhi Society, Chembur, Mumbai, Maharashtra - 400 071.**

Accredited by NAAC "A Grade" in 3rd Cycle-2017

**Best College Award- Urban Area, University of
Mumbai (2012-13) Recipient of FIST Grant
(DST) and STAR College Grant (DBT)**

Credit Based System for Holistic Development Syllabus

For

F. Y. B. Sc. Artificial Intelligence

Undergraduate Programme

As per NEP 2020

With effect from academic year 2026-27

Course Structure

B. Sc. (Artificial Intelligence) Programme

B. Sc. (Artificial Intelligence) is a 3/4 year undergraduate course designed to provide theoretical and practical knowledge of Artificial Intelligence along-with core subjects in the field of computer science and information technology. In order to master the skills, a learner should have strong mathematical, statistical and computational skills. The course aims to cover all the important subjects required for the same.

Course Highlights

Name of the Course	B.Sc. (Artificial Intelligence)
Level	Undergraduate
Duration	3 /4 Years
Minimum Academic Requirement	10 + 2
Minimum Aggregate Score Requirement	Mathematics as compulsory subject in 12th
Stream Requirement	XII Science / Commerce / Arts with Mathematics subject (Merit List on Mathematics marks)
Course mode	Full – Time
Exam Type	Semester

(AI)Program Objectives:

1. The objective of studying Artificial Intelligence at the graduation level is to help students understand the fundamental concepts and techniques used to build intelligent systems.
2. It enables learners to develop problem-solving skills by applying algorithms, data analysis, and logical thinking to real-world engineering challenges.
3. Students gain knowledge of machine learning, data handling, and automation, which are essential in modern industries.
4. The course also encourages innovation by allowing students to design smart applications and solutions.
5. It prepares them to use AI tools effectively for improving productivity and decision-making.
6. Ethical considerations and responsible use of AI are also emphasized. Students become

- capable of adapting to rapidly evolving technologies and industry demands.
7. Overall, it builds a strong foundation for careers, higher studies, and entrepreneurship in AI-related fields.

(AI)Program Outcomes:

PO1: "Basic and Discipline specific knowledge:

Apply knowledge of basic mathematics, science and engineering fundamentals and engineering specialization to solve the engineering problems"

PO2: Problem analysis: Identify and analyse well-defined engineering problems using codified standard methods.

PO3: "Design/ development of solutions: Design solutions for well-defined technical problems and assist with the design of systems components or processes to meet specified needs."

PO4: "Engineering Tools, Experimentation and Testing: Apply modern engineering tools and appropriate technique to conduct standard tests and measurements."

PO5: "Engineering practices for society, sustainability and environment: Apply appropriate technology in context of society, sustainability, environment and ethical practices. "

PO6: "Project Management: Use engineering management principles individually, as a team member or a leader to manage projects and effectively communicate about well-defined engineering activities. "

PO7: Life-long learning: Ability to analyse individual needs and engage in updating in the context of technological changes.

VES COLLEGE OF ARTS, SCIENCE AND COMMERCE

BSC (ARTIFICIAL INTELLIGENCE)

First YEAR NEW SYLLABUS (2025-2026)

SEMESTER 1

Course	Sem	Course Code	Sr. No.	Name of Course	Type of Course	Credit	Subject Type	External Exam Marks	Internal Exam Marks	Total Marks
BSc- AI	I		1	C Programming	MAJOR	2	THEORY	30	20	50
	I		2	Operating Systems	MAJOR	2	THEORY	30	20	50
	I		3	Practical of (C Programming and Operating Systems)	MAJOR	2	PRACTICAL	-	50	50
	I		4	Vedic Maths	IKS	2	THEORY	30	20	50
	I		5	Introduction to Calculus in AI	VSC	2	THEORY	30	20	50
	I		6	Web Page Designing (CSS,Html,JAVA script)	SEC	2	PRACTICAL	-	50	50
	I		7	Business Communication - I	AEC	2	THEORY	30	20	50
	I		8	Environmental Science -I	VEC	2	THEORY	30	20	50
	I		9	OE-1	OE	2	THEORY	30	20	50
	I		10	OE-2	OE	2	THEORY	30	20	50
	I		11	Co-Curricular Courses	CC	2	-	-	-	50
						22				

VES COLLEGE OF ARTS, SCIENCE AND COMMERCE

BSC (ARTIFICIAL INTELLIGENCE)

First YEAR NEW SYLLABUS (2025-2026)

SEMESTER 2

Course	Sem	Course Code	Sr. No.	Name of Course	Type of Course	Credit	Subject Type	External Exam Marks	Internal Exam Marks	Total Marks
BSc- AI	II		1	Python Programming	MAJOR	2	THEORY	30	20	50
	II		2	Relational Database Management Systems(RDBMS)	MAJOR	2	THEORY	30	20	50
	II		3	Practical on (Python Programming & RDBMS)	MAJOR	2	PRACTICAL	-	50	50
	II		4	Discrete Mathematics & Linear Algebra	MAJOR	2	THEORY	30	20	50
	II		5	AI based User Interface Design	MINOR	2	PRACTICAL	-	-	50
	II		6	OOP Using Java	MINOR	2	PRACTICAL	-	-	50
	II		7	Environmental Science - II	VEC	2	THEORY	30	20	50
	II		8	Business Communication-II	AEC	2	THEORY	30	20	50
	II		9	OE-1	OE	2	THEORY	30	20	50

	II		10	OE-2	OE	2	THEORY	30	20	50
	II		11	Co-Curricular Courses	CC	2	-	-	-	50
						22				

SEMESTER

ONE

F. Y. B. Sc. Artificial Intelligence
SEMESTER I
Major / Core Course I
COURSE TITLE: C Programming
[CREDITS - 02]

Course Learning Objective

The course is designed to provide complete knowledge of C language. Students will be able to develop logics which will help them to create programs, applications in C. Also by learning the basic programming constructs they can easily switch over to any other language in future. After the completion of this course, the students will be able to develop applications.

Course Learning Outcome

After learning this course, students will be able to

CO1: Apply fundamental concepts of C programming, including data types, operators, input/output, algorithms, and flowcharts, to develop simple problem-solving programs.

CO2: Develop C programs using decision-making, looping constructs, and arrays to solve structured and repetitive computational problems.

CO3: Design and implement modular C programs using structures, functions, and pointers for efficient data handling and problem-solving.

Module No.	Name Of The Module	Lectures
1	Unit I : Basics of C Programming 1.1 Introduction to C: History of 'C' General Structure of a 'C' program: Header files, 'main' function. 1.2 Data Concepts: Character set, tokens, keywords, Identifiers, Variables, Constant, data types, C operators, Arithmetic operators, Arithmetic expression, declaring variables, and data type conversion. 1.3 Basic Input output: printf() and scanf(), character input/output statements, Input/output formatting, Use of comments. 1.4 Fundamentals of algorithms: Notion of an algorithm. Pseudo-code conventions like assignment statements and basic control structures. 1.5 Operators: Arithmetic, relational, logical, bitwise, assignment 1.6 Algorithmic problems: Develop fundamental algorithms to solve simple problems 1.7 Flowchart: Symbols of flowchart, Guidelines for preparing Flowchart	10
2	Unit II : Control Structures and Array in C	10

	2.1 Decision making and branching: Relational and logical operators, if statement, if else statement, nested if- else, if-else ladder' The switch statement 2.2 Looping: While loop, Do... While loop For loop, Go to statement, Use of break and continue statements 2.3 Array-Characteristics of an array, Types of Array- One dimension and two dimensional arrays 2.4 Array declaration and Initialization	
3	Unit III : Structures, Functions and Pointers in C 3.1 Introduction and Features of Structures, Declaration and Initialization of Structures Type def, Enumerated Data Type, using structures in C Program 3.2 Concept and need of functions, Library functions: Math functions, String handling functions, other miscellaneous functions. 3.3 Writing User defined functions, scope of variables. 3.4 Parameter passing: call by value, call by reference. Recursive functions 3.5 Concepts of pointers: declaring, initializing, accessing, Pointer arithmetic. 3.6 Handling arrays using pointers	10
Reference	Textbooks 1.Let us 'C', Kanetkar , BPB 2.Programming in ANSI, Balgurusamy, Tata Mc-Graw Hill 3.C for beginners, Madhusudan Mothe, SPD Additional Reference 1.www.w3schools.com, www.javatpoint.com 2.https://www.cprogramming.com/tutorial.html	

Modality of Assessment

The performance of the learners shall be evaluated into two parts. The learner's performance shall be assessed by Internal Assessment with 20 marks in the first part & by conducting the Semester End Examinations with 30 marks in the second part.

Students will have to score 40% of marks in the Internal Assessment as well as Semester End Examination to pass the course.

Internal Assessment: It is defined as the assessment of the learners on the basis of internal evaluation as envisaged in the Credit & Choice Based System by way of participation of learners in various academic and correlated activities in the given semester of the programme.

Semester End Assessment: It is defined as the assessment of the learners on the basis of Performance in the semester end Theory/ Written examination.

The allocation of marks for the Internal Assessment and Semester End Examinations are as shown below:-

Evaluation Pattern

Evaluation Pattern Credits -2

Continuous Internal Evaluation Out of 20 marks

Semester End Evaluation Out of 30 marks

Total Marks 50 marks

Continuous Internal Evaluation

- **Class Test: 10 marks**
- **Assignment / Group presentation /Quiz/Project/Role play/Environmental skits/Viva-voce : 10 marks**

Question Paper Pattern

Maximum Marks: 30

Questions to be set: 03

Duration: 75 mins

Note: All questions are compulsory carrying 10 marks each.

Q. No.	Particulars	Marks
1	Answer the Following (Any Two) A. B. C.	05 05 05

2	Answer the Following (Any Two)	05
	A.	05
	B.	05
	C.	
3	Answer the Following (Any Two)	05
	A.	05
	B.	05
	C.	
Total		30

F. Y. B. Sc. Artificial Intelligence

SEMESTER I

Major / Core Course II

COURSE TITLE: Operating Systems

[CREDITS - 02]

Course Learning Objective

This course is introduced to make aware of different types of Operating System and their services. It includes of core structure, functions and design principles of operating system scheduling algorithms. File management and memory management. Operating System is Important for Computer Systems as it makes us understand the structure and organization of the file system, what a process is and how processes are synchronized and scheduled and different approaches to memory management. After learning this course students will be able to use system calls for managing processes, memory and the file system.

Course Learning Outcome

CO1.Install operating system, configure it and study evaluation of it
CO2.Execute process commands for performing process management operations
CO3.Calculate efficiency of different memory management techniques and study different algorithms for CPU Process Scheduling

Module No.	Name Of The Module	Lectures
1	Unit I: Fundamentals of Operating system 1.1 Introduction to Operating system and functions of an operating system. 1.2 Structure of operating system 1.3 Views of operating system- User view, System View	10

	1.4 Types of operating systems- Batch operating system, Multi Programmed, Time Shared OS, Multiprocessor Systems, Distributed Systems, Real time systems. Mobile OS 1.5 Command line based OS – DOS, UNIX GUI based OS –WINDOWS, LINUX 1.6 Different Services of Operating System 1.7 System Calls- Concept, types of system calls	
2	Unit II: Process & Thread Management 2.1 Program vs. Process 2.2 Process control block(PCB) 2.3 State transition diagram 2.4 Scheduling Queues and context switch 2.5 Scheduler and Types of schedulers 2.6 Inter-process communication (IPC): Introduction, shared memory system and message passing system 2.7 Threads - Benefits, users and kernel threads, Multithreading Models - Many to One, One to One, Many to Many	10
3	Unit III: CPU scheduling and memory management 3.1 CPU scheduling and it's need 3.2 Types of scheduling- Pre-emptive, Non- Pre-emptive Scheduling 3.3 Scheduling criteria 3.4 Scheduling algorithms-(FCFS, SJF, Round-Robin, Priority and Multilevel queue scheduling) 3.5 Memory management- Partitioning, static and dynamic 3.6 Virtual Memory – Introduction to Paging, Segmentation and Fragmentation 3.7 Page replacement algorithms- FIFO, LRU, Optimal. 3.8 File – Concepts, Attributes, Operations, types and File System Structure, Access Methods – Sequential, Direct, File Allocation Methods- Contiguous, Linked, Indexed 3.9 Directory structure— Single level, two levels, tree-structured directory, Disk Organization and disk Structure- Physical structure, Logical structure, RAID structure of disk, RAID level 0 to 6	10
References	Textbooks 1.Operating Systems Concepts, 8th Edition, A Silberschatz, P.B. Galvin, G. Gagne, John Wiley Publications 2008 2. Modern Operating SystemsA.S. Tanenbaum, Pearson Education 2007. 3Operating Systems: Internals and Design Principles, Seventh Edition, William Stallings, Prentice Hall. Copyright © 2012 by Pearson Education. Additional References 1.www.guru99.com 2.www.tutorialspoint.com	

	3. www.javapoint.com 4. www.scaler.com 5. www.geeksforgeeks.org	
--	---	--

Modality of Assessment

The performance of the learners shall be evaluated into two parts. The learner's performance shall be assessed by Internal Assessment with 20 marks in the first part & by conducting the Semester End Examinations with 30 marks in the second part.

Students will have to score 40% of marks in the Internal Assessment as well as Semester End Examination to pass the course.

Internal Assessment: It is defined as the assessment of the learners on the basis of internal evaluation as envisaged in the Credit & Choice Based System by way of participation of learners in various academic and correlated activities in the given semester of the programme.

Semester End Assessment: It is defined as the assessment of the learners on the basis of Performance in the semester end Theory/ Written examination.

The allocation of marks for the Internal Assessment and Semester End Examinations are as shown below:-

Evaluation Pattern

Evaluation Pattern Credits -2

Continuous Internal Evaluation Out of 20 marks

Semester End Evaluation Out of 30 marks

Total Marks 50 marks

Continuous Internal Evaluation

- **Class Test: 10 marks**
- **Assignment / Group presentation /Quiz/Project/Role play/Environmental skits/Viva-voce : 10 marks**

Question Paper Pattern

Maximum Marks: 30

Questions to be set: 03

Duration: 75 mins

Note: All questions are compulsory carrying 10 marks each.

Q. No.	Particulars	Marks
1	Answer the Following (Any Two) A. B. C.	05 05 05
2	Answer the Following (Any Two) A. B. C.	05 05 05
3	Answer the Following (Any Two) A. B. C.	05 05 05
Total		30

F. Y. B. Sc. Artificial Intelligence

SEMESTER I

Major / Core Course III

COURSE TITLE: Practical of C Programming and Operating System

[CREDITS - 02]

Sr no	Title of Practical of C Programming
1	To draw flowchart and write algorithms for sample program
2	Write a program using constant ,variable and arithmetic expression
3	To write a C program for various operators in 'C'
4	Write a program using if, if..else statement
5	Write a program using while, do..while
6	Write a program using switch statement
7	To write a C program for single dimensional integer arrays
8	Develop a program using structure
9	Develop a program using user define function
10	To write a C program for pointers to print values of variables and their addresses
11	To write a C program to demonstrate the concept of pointer arithmetic.
12	Develop a program for file handling

Sr no	Title of Practical of Operating Systems
1	Install and configure Linux (or alike) operating system
2	Execute file and directory manipulation commands using linux
3	Apply access permission commands on files or folders using linux
4	Write program to calculate sum of n numbers using thread library
5	Write program to implement FCFS algorithm.
6	Write program to implement Round Robin scheduling algorithm.
7	Write program to implement SJF scheduling algorithm
8	Write program to implement non-preemptive priority based scheduling algorithm
9	Write program to implement preemptive priority based scheduling algorithm
10	Write program to implement SRJF scheduling algorithm
11	Write a program to implement page replacement algorithm FIFO/LRU
12	Write a program to organize the file using single level directory

F. Y. B. Sc. Artificial Intelligence
SEMESTER I
Vocation Skill Course VSC 1
COURSE TITLE: Introduction to Calculus
[CREDITS - 02]

Course Learning Objective

In calculus, we use three main tools for analyzing and describing the behavior of functions: limits, derivatives, and integrals. Students will use these tools to solve application problems in a variety of settings ranging from physics and biology to business and economics. Compute limits, derivatives, and integrals

Course Learning Outcome

CO1: *Understand and apply* the concepts of functions, limits, and derivatives to *analyze* and *solve* problems related to rates of change, tangents, normals, maxima, minima, and curvature.
CO2: *Apply and analyze* various methods of integration and properties of definite integrals to *evaluate* areas, volumes, and solve practical problems involving accumulation.
CO3: *Understand and apply* different representations of complex numbers and perform algebraic operations to *solve* mathematical and engineering problems.

Module No.	Name Of The Module	Lectures
1	Unit I : Differential calculus and Applications of Derivatives 1.1 Definition of Functions and Types of Functions with Examples 1.2 Concept of Limits with Examples 1.3 Derivatives 1a] Rules of Derivatives 2a] Chain Rule 1.4 Derivatives of Composite Functions 1.5 Applications of Derivatives[equation of tangent and normal, maxima and minima, radius of curvature]	
2	Unit II : Integral calculus 2.1 Rules of integrations 2.2 Methods of integrations [substitution method, integration by parts, partial fraction method] 2.3 Definite integral 2.4 Properties of definite integral 2.5 Area under one curve and between two curves 2.6 Volume generated by revolution of curves.	
3	Unit III : Complex number 3.1 Definition of complex no. 3.2 Cartesian ,polar and exponential form of complex no.with examples 3.3 Algebra of complex no.with examples	

Reference	Textbooks 1.Higher Engineering Mathematics, B. S. Grewal, Khanna Publishers; Forty Fourth edition (1 January 1965) 2.Advanced Engineering Mathematics, Kreyszig Erwin, Wiley India Pvt. Ltd 3.Advanced Engineering Mathematics, H K Dass, S Chand Additional References 1.https://www.youtube.com/watch?v=L6_c6qvlB8I&list=PLzJaFd3A7DZuyLLbmVpb9e9VLf3Q9cYBL 2.https://www.khanacademy.org/math/calculus-1	
------------------	--	--

Modality of Assessment

The performance of the learners shall be evaluated into two parts. The learner's performance shall be assessed by Internal Assessment with 20 marks in the first part & by conducting the Semester End Examinations with 30 marks in the second part.

Students will have to score 40% of marks in the Internal Assessment as well as Semester End Examination to pass the course.

Internal Assessment: It is defined as the assessment of the learners on the basis of internal evaluation as envisaged in the Credit & Choice Based System by way of participation of learners in various academic and correlated activities in the given semester of the programme.

Semester End Assessment: It is defined as the assessment of the learners on the basis of Performance in the semester end Theory/ Written examination.

The allocation of marks for the Internal Assessment and Semester End Examinations are as shown below:-

Evaluation Pattern

Evaluation Pattern Credits -2

Continuous Internal Evaluation Out of 20 marks

Semester End Evaluation Out of 30 marks

Total Marks 50 marks

Continuous Internal Evaluation

- **Class Test: 10 marks**

- **Assignment / Group presentation /Quiz/Project/Role play/Environmental skits/Viva-voce : 10 marks**

Question Paper Pattern

Maximum Marks: 30

Questions to be set: 03

Duration: 75 mins

Note: All questions are compulsory carrying 10 marks each.

Q. No.	Particulars	Marks
1	Answer the Following (Any Two) A. B. C.	05 05 05
2	Answer the Following (Any Two) A. B. C.	05 05 05

3	Answer the Following (Any Two)	05
	A.	05
	B.	05
	C.	
Total		30

F. Y. B. Sc. Artificial Intelligence

SEMESTER I

Skill Enhancement Course

Course Title: Web Page Designing

[CREDITS -02]

Course Learning Objective

The objective of this course is to provide a strong foundation in web page design by enabling students to understand and apply HTML, CSS, and JavaScript for creating interactive and responsive web pages. It aims to develop skills in planning and conducting user research, integrating social media content, and designing user-friendly and accessible websites. The course also focuses on applying basic design and programming principles, along with effective website project management techniques to build functional and visually appealing web solutions.

Course Learning Outcomes

CO1: Understand and apply HTML concepts, tags, and attributes to design web pages, and demonstrate the process of website publishing and hosting on intranet and internet environments.

CO2: Apply and analyze CSS techniques, including styling, layouts, and responsive design, to create visually appealing and user-friendly web interfaces.

CO3: Apply and evaluate JavaScript concepts, DOM manipulation, and browser features (including cookies) to develop interactive and dynamic web applications.

Module	Name Of The Module	30 L
--------	--------------------	------

1.	HTML 1.1 Basic HTML (Introduction of HTML, HTML Basic Tags and Attributes, Hyperlinks, Physical Style Tags, Images, Tables and Lists, Multimedia Objects) 1.2 Advance HTML (Frames, Forms, Image Mapping, Embedding Multimedia, Applets, Web Server, Browsers) 1.3 Website Hosting: Concept of Internet and Intranet. Publishing website on Intranet, installing and configuring web server, uploading files on intranet site, accessing intranet based website, publishing website site on the Internet, hiring web space, uploading files using FTP, virtual hosting, accessing internet based website.	10
2.	CSS 2.1 Cascading Style Sheets: Different types of Style Sheets and the benefits of using CSS. Adding style to the document: Linking to style sheets, embedding style sheets, using inline style, and selecting the class rules, and ID rules. 2.2 Style sheet properties: Font, text, box, color, and background properties; Creating and Using a simple external CSS file; Using the internal and inline CSS; background and color gradients in CSS Setting font and text in style sheet using table layout. 2.3 CSS Advanced: CSS Rounded Corners, CSS Border Images, CSS Shadows, CSS Text Effects, CSS 2D Transforms, CSS 3D Transforms, CSS Transitions, CSS Animations, CSS Tooltips, CSS Style Images, CSS Image Reflection.	10
3.	JavaScript 3.1 Introduction to JavaScript (Datatypes, Variables, Operators, Array, String, Functions, Conditional Statements, Loop Structure) 3.2 JavaScript Object Model (DOM, Form Properties and Methods, Form Events, Form Objects, Creating Responsive Menu, Creating Password Generator, Creating To Do List, Creating JavaScript Calculator, Creating Date Object) 3.3 Cookies (Basics of Cookie, Reading and Writing Cookie Value, Creating, Deleting, Updating Cookies, Opening a Window, Giving a new Window focus, Window Positioning)	10

List of Practical to be conducted

Practical No.	Practical Name
1.	Create a structured web page using <header>, <nav>, <section>, <aside>, and <footer>.
2.	Use heading tags (<h1> to <h6>) dynamically to create a collapsible FAQ section.

3.	Style and format multiple paragraphs using different font families, text shadows, and spacing.
4.	Use block-level and inline elements together to create a multi-column blog layout.
5.	Mix text-level and block-level elements to create an article layout with citations and highlights.
6.	Create a navigation menu using an ordered/unordered list with dropdowns.
7.	Design a custom bullet list using images or CSS properties.
8.	Create a glossary page using special characters, tooltips, and term definitions.
9.	Develop a multi-page website where each page links to the next using navigation buttons.
10.	Style links with different hover effects (color changes, animations, etc.).
11.	Use an image as a hyperlink and add a hover effect that changes the image.
12.	Create a breadcrumb navigation system using hyperlinks.
13.	Insert a background image that changes dynamically based on the time of day.
14.	Use foreground and background images together with overlay effects.
15.	Create an interactive image gallery with clickable thumbnails and full-size views.
16.	Design an advanced table layout with merged cells, images, and custom styles.
17.	Create a pricing table with highlighted recommended plans and hover effects.
18.	Develop a data table with sorting functionality using HTML and basic JavaScript.
19.	Use an HTML iframe to embed a YouTube video and customize its display settings.
20.	Create a multi-frame layout where one frame acts as a navigation bar for others.
21.	Build a simple portfolio page using <code><section></code> , <code><article></code> , and <code><aside></code> .
22.	Apply CSS to customize all elements (font, spacing, layout, colors) for a clean design.
23.	Use a responsive grid layout with media queries to ensure mobile compatibility.
24.	Host a basic portfolio website on GitHub Pages or another open-source platform.

SEMESTER TWO

F. Y. B. Sc. Artificial Intelligence

SEMESTER II

Major / Core Course I

COURSE TITLE: Python Programming

[CREDITS - 02]

Course Learning Objective

The objective of this course is :

This course imparts the basic concepts of data structures and algorithms, searching and sorting techniques, basic concepts about stacks, queues, lists, trees and graphs. Understanding about writing algorithms and step by step approach in solving problems with the help of fundamental data structures. This makes easy for users to access and work with the data they need in appropriate ways. Most importantly, data structures frame the organization of information so that machines and humans can better understand it. Learning this subject students will gain the ability to implement relevant algorithms using Data Structures.

Course Learning Outcomes

After completion of this course learner will be able to:

CO1: Understand and apply fundamental Python concepts such as syntax, data types, variables, and control flow statements to develop basic programs.

CO2: Apply and analyze functions, file handling, exception handling, and object-oriented programming concepts to design modular and error-handling Python applications.

CO3: Apply and evaluate Python data structures and libraries to efficiently store, manipulate, and process data in problem-solving scenarios.

Module	1. Name of Module	Lectures
1	Introduction to Python 1.1 Python Installations, IDE Setup, Basic Syntax, Comments	15

	1.2Data Types and Variable: Data Types (int, float, etc), Variables, Typecasting. 1.3Control Flow: Conditional Statements like (if, else, elif), loops (for loop, while loop)	
2	Functions and Modules 2.1Defining Functions Parameters, Return Values, Importing Modules 2.2File Handling: Reading from files, writing to files, files methods 2.3Exception Handling: Try/Except Blocks, Raising Exceptions Finally Cause 2.4Introduction to Object Oriented Programming: Classes and Objects, Attributes and Methods, Inheritance.	15
3	Introduction To Data Structure 3.1 Overview of Data Structure, Lists, Tuples 3.2 Working with Libraries: Using Standard Libraries (math, datetime), 3.3 Introduction to third party libraries 3.4 Advance Data Structures: Dictionaries, Sets, String Manipulation.	15
Referenc e	Textbooks: 1. Data Structures using C, Balagurusamy, McGraw Hill Education. 2. Computer Algorithms, Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, algotia Publication. 3. Data Structures using C, Li Lipschutz, McGraw Hill Education Additional References: 1. www.geeksforgeeks.org 2. www.javapoint.com 3. www.w3schools.com	

Modality of Assessment

The performance of the learners shall be evaluated into two parts. The learner's performance shall be assessed by Internal Assessment with 20 marks in the first part & by conducting the Semester End Examinations with 30 marks in the second part.

Students will have to score 40% of marks in the Internal Assessment as well as Semester End Examination to pass the course.

Internal Assessment: It is defined as the assessment of the learners on the basis of internal

evaluation as envisaged in the Credit & Choice Based System by way of participation of learners in various academic and correlated activities in the given semester of the programme.

Semester End Assessment: It is defined as the assessment of the learners on the basis of Performance in the semester end Theory/ Written examination.

The allocation of marks for the Internal Assessment and Semester End Examinations are as shown below:-

Evaluation Pattern

Evaluation Pattern Credits -2

Continuous Internal Evaluation Out of 20 marks

Semester End Evaluation Out of 30 marks

Total Marks 50 marks

Continuous Internal Evaluation

- **Class Test: 10 marks**

- **Assignment / Group presentation /Quiz/Project/Role play/Environmental skits/Viva-voce : 10 marks**

Question Paper Pattern

Maximum Marks: 30

Questions to be set: 03

Duration: 75 mins

Note: All questions are compulsory carrying 10 marks each.

Q. No.	Particulars	Marks
1	Answer the Following (Any Two) A. B. C.	05 05 05
2	Answer the Following (Any Two) A. B. C.	05 05 05
3	Answer the Following (Any Two) A. B. C.	05 05 05
Total		30

F. Y. B. Sc. Artificial Intelligence

SEMESTER II

Major / Core Course II

COURSE TITLE: Relational Database Management System

[CREDITS – 02]

Course Learning Objectives

This course focuses on fundamentals of relational database management system and enables students to design and manage database for various software applications. It also provides students with theoretical knowledge and practical skills in the use of databases and database management systems in Information Technology applications.

Course Learning Outcomes

- 1.Explain concept of database management system.
2. Design the database for given problem.
- 3.Manage database using SQL.

Module No.	Name Of The Module	Lectures
1	Unit 1 : Database System Concepts 1.1 Concept of Data, Database, DBMS, Advantages of DBMS over file system, Applications of DBMS 1.2 Three level Architecture for Database 1.3 Data Abstraction : Differents levels of Data Abstraction, Instance and Schema, Data Independence – Logical and Physical Data Independence 1.4 Overall Structure of DBMS	
2	Unit II : Data Modeling , Normalization & Relational Model, Relational Calculus 2.1 Data Modeling : Record based Logical model- Relational, Network,Hierarchical, ER Model: Entity Relationship Model, Strong Entity set, Weak Entity Set 2.2 Fundamentals Of RDBMS- Record, Fields, Datatypes, tables and Databases. 2.3 Concept of Keys in DBMS, Candidate key, Primary Key, Foreign Key 2.4 Normalization, concepts, Need of normalization, Types of Normalization. Types of Normalization. 1NF, 2NF, 3NF.	
3	Unit III : Create and Manage Database using SQL Command. 3.1 Introduction to Structured Query Language, Data types of SQL, Components of SQL-DDL, DML. DCL, DQL 3.2 DDL Commands: CREATE, ALTER, DROP, TRUNCATE,DESC,RENAME. 3.3 Data Integrity Constraints: Types Of Data Integrity Constraint: I/O constraint-Primary Key, Foreign key, Unique key constraint, Business Rule Constraint-Null, Not Null and Check Constraint. 3.4 DML commands : INSERT , UPDATE , DELETE 3.5 DCL commands : COMMIT , SAVEPOINT , ROLLBACK, GRANT and REVOKE. 3.6 DQL commands : SELECT.	

	3.7 SQL Operators: Arithmetic Operators, Comparison Operator, Logical Operators , Set Operators, Range Searching Operators- Between, Pattern matching operators-LIKE 3.8 In-built Functions: String , Arithmetic, Date and Time function, Aggregate Functions. 3.9 Queries using Group by having and order by clause, Joins- Inner and Outer Join, Sub-Queries. 3.10 Views: Concepts of view, The Create view command, Updating views, Views and Joins, views and Sub Queries, Dropping Views, Sequences: Creating sequences, Altering Sequences, Dropping Sequences, 3.11 Indexes: Types, Creating an simple Indexes, Simple, Unique and Dropping Indexes. 3.12 Synonyms: Creating Synonyms, Dropping Synonymns.	
Reference	Textbooks: 1.Introductions to database management system ,ISRD Group, McGraw Hill Education. 2.Introduction to Relational Database & SQL Programming, Allen, McGraw Hill Education. 3.Database System Concepts , Korth, Mcgraw Hill Education.	

Modality of Assessment

The performance of the learners shall be evaluated into two parts. The learner's performance shall be assessed by Internal Assessment with 20 marks in the first part & by conducting the Semester End Examinations with 30 marks in the second part.

Students will have to score 40% of marks in the Internal Assessment as well as Semester End Examination to pass the course.

Internal Assessment: It is defined as the assessment of the learners on the basis of internal evaluation as envisaged in the Credit & Choice Based System by way of participation of learners in various academic and correlated activities in the given semester of the programme.

Semester End Assessment: It is defined as the assessment of the learners on the basis of Performance in the semester end Theory/ Written examination.

The allocation of marks for the Internal Assessment and Semester End Examinations are as shown below:-

Evaluation Pattern

Evaluation Pattern Credits -2

Continuous Internal Evaluation Out of 20 marks

Semester End Evaluation Out of 30 marks

Total Marks 50 marks

Continuous Internal Evaluation

- **Class Test: 10 marks**

- **Assignment / Group presentation /Quiz/Project/Role play/Environmental skits/Viva-voce : 10 marks**

Question Paper Pattern

Maximum Marks: 30

Questions to be set: 03

Duration: 75 mins

Note: All questions are compulsory carrying 10 marks each.

Q. No.	Particulars	Marks

1	Answer the Following (Any Two) A. B. C.	05 05 05
2	Answer the Following (Any Two) A. B. C.	05 05 05
3	Answer the Following (Any Two) A. B. C.	05 05 05
Total		30

F. Y. B. Sc. Artificial Intelligence

SEMESTER II

Major / Core Course II

COURSE TITLE: Practical of Python and RDBMS

[CREDITS - 02]

Course Code	Course Title	Credits
	Practical of Python Programming	1
1	Install a Python IDE and Display a Simple Message.	
2	Write a Python program to implement Basic Operators in python.	

3	Write a program to check whether an input number is positive , negative or zero using a conditional statement.	
4	Write a Program to illustrates Loop case statement	
5	Write a program to demonstrate various predefined & user defined functions	
6	Write a Program Using built-in math functions in Python.	
7	Write a Program Using Built in Mathematics Modules in Python.	
8	Strings (Conversion of hexadecimal Number into its Equivalent Binary number).	
9	List processing searching and sorting (sorting based on the length of each element.)	
10	Write Python program to perform following operations on lists, Tuples , Sets , Dictionaries : a) Create b) Access c) Update d) Delete elements in list	
11	Write a program to multiply two matrices using numpy	
12	Write a program to create user defined packages in Python.	
13	Implement the concept of polymorphism in Python.	
14	Develop program in Python to demonstrate following operations: a) Single inheritance b) Multilevel inheritance c) Multiple inheritance d) Hybrid inheritance e) Hierarchical inheritance	
15	Write a program in Python for handling array to demonstrate following operations : a) Array declaration b) Insertion c) Deletion d) Append e) Index f) Reverse	
16	Write a program in Python for linked list to demonstrate following operations : a) Insertion b) Deletion c) Updating d) Merging to list.	
17	Write a program in Python to demonstrate queue data structure operations : a) Enqueue b) Dequeue c) Display	

18	Write a program to implement Binary Search Tree.	
19	Write a program for Linear Search and Binary search.	
20	Write a program to implement Bubble Sort and Selection Sort	
21	Write a program to implement Merge sort and Quick sort.	
22	Write a program to implement Stacks and Queues	
23	Write a program to implement a Singly Linked List. Write a program to implement Doubly Linked list	
24	Write a program to implement Exponential Search Interpolation Search	

Relational Database Management System (RDBMS)		
Course Code	Course Title	Credits
	Practical of Relational Database Management System (RDBMS).	
1	Perform following in GUI based software using GUI only i) Create Database ii) Create tables and assign primary key. iii) Modify the table structure-add column change the data type of column, delete the record from table.	
2	Design E-R diagram and Create Normalized Database on the given data.	
3	i) Create and Execute DDL commands using SQL. ii) Apply following integrity constraints on table. iii) Primary key, Foreign key, Unique key constraint, Null, Not Null, and check constraint.	
4	Create and Execute DML commands using SQL.	
5	Write Queries using operators: Arithmetic Operators, Comparison Operator, Logical Operator, Set Operators, Range Searching Operators-Between, Pattern Matching Operators-Like.	
6	Write Queries using following Functions: String, Arithmetic, Date and Time, Aggregate Functions.	
7	Execute Queries using the select command with Where, Having, Group by and order by clause.	
8	Execute Queries for implementation of Inner and Outer join.	

9	Implement views i)create different views.	
10	Create and Execute Indexes, Sequences and synonyms in SQL.	
11	Make a database for any institute, company etc. and demonstrate use of DDL, DML apply different functions.	
12	Make a database for any institute, company etc. and demonstrate use of DDL, DML apply use of different operator.	

F. Y. B. Sc. Artificial Intelligence

SEMESTER II

Minor Course I

COURSE TITLE: Discrete Mathematics and Linear Algebra

[CREDITS - 02]

Course Learning Objective

The objective of this course is :

The mathematics of modern day computing world is built around discrete mathematics. Discrete mathematics will help us to deal with the algorithm complexity. The understanding of matrices, set theory, along with discrete mathematics will help students to analyse the algorithms by successfully identifying its parameters, capability and limitations.

Course Learning Outcome

After completion of this course learner will be able to:

CO1: *Apply and analyze* matrix operations, determinants, and eigenvalues to solve systems of linear equations and related algebraic problems.

CO2: *Understand and apply* principles of logic and combinatorics, including propositional and predicate calculus, counting techniques, and generating functions to solve discrete mathematical problems.

CO3: *Understand and analyze* set theory concepts, relations, functions, and Boolean algebra to model and solve problems in discrete structures.

Module No.	Name Of The Module	Lectures
1	Matrices 1.1 Definition and types of matrix 1.2 Algebra of matrices 1.3 Properties of matrices 1.4 Determinants of matrix 1.5 Inverse of matrix by adjoint method and by elementary transformation 1.6 Rank of matrix. 1.7 System of linear equation using inverse matrix. 1.8 Solution of system of linear equations by LU decomposition method 1.9 Eigen values, Eigen vectors 1.10 Cayley hamilton theorem	10

2	Mathematical Logic & Combinatorics 2.1 First Order Logic 2.2 Logical Operators 3.3 Well-Formed formulas 2.4 Propositional Calculus 2.5 Predicate Calculus 2.6 Universal Quantifiers & Existential Quantifiers principles of counting 2.7 Permutations and Combinations 2.8 Generating functions 2.9 Binomial coefficients	10
3	Set Theory and Boolean Algebra 3.1 Set theory 3.2 Set Operations 3.3 Relations and their types 3.4 Functions and their types 3.5 Groups, Lattice, Posets 5.6 Boolean Algebra	10
References	Textbooks: 1. Higher Engineering Mathematics, B. S. Grewal, Khanna Publishers; Forty Fourth edition (1 January 1965) 2. Advanced Engineering Mathematics, Kreyszig, Erwin Wiley India Pvt. Ltd 3. Advanced Engineering Mathematics, H K Dass, S Chand Additional References:	

Modality of Assessment

The performance of the learners shall be evaluated into two parts. The learner's performance shall be assessed by Internal Assessment with 20 marks in the first part & by conducting the Semester End Examinations with 30 marks in the second part.

Students will have to score 40% of marks in the Internal Assessment as well as Semester End Examination to pass the course.

Internal Assessment: It is defined as the assessment of the learners on the basis of internal evaluation as envisaged in the Credit & Choice Based System by way of participation of learners in various academic and correlated activities in the given semester of the programme.

Semester End Assessment: It is defined as the assessment of the learners on the basis of Performance in the semester end Theory/ Written examination.

The allocation of marks for the Internal Assessment and Semester End Examinations are as shown below:-

Evaluation Pattern

Evaluation Pattern Credits -2

Continuous Internal Evaluation Out of 20 marks

Semester End Evaluation Out of 30 marks

Total Marks 50 marks

Continuous Internal Evaluation

- **Class Test: 10 marks**
- **Assignment / Group presentation /Quiz/Project/Role play/Environmental skits/Viva-voce : 10 marks**

Question Paper Pattern

Maximum Marks: 30

Questions to be set: 03

Duration: 75 mins

Note: All questions are compulsory carrying 10 marks each.

Q. No.	Particulars	Marks
1	Answer the Following (Any Two) A. B. C.	05 05 05

2	Answer the Following (Any Two) A. B. C.	05 05 05
3	Answer the Following (Any Two) A. B. C.	05 05 05
Total		30

F. Y. B. Sc. Artificial Intelligence

SEMESTER II

Minor Course- II

COURSE TITLE: AI Based User Interface Design

COURSE CODE:

[CREDITS - 02]

Course Learning Outcomes(chk sequence of objective and outcome)		
After completion of this course learner will be able to: CO1.Think critically on product design concepts with respect to user defined requirements. CO2. Create interactions using design tools and use appropriate visual design for given problems. CO3. Develop a prototype for given applications.		
Course Learning Objective		
The objective of this course ,UI/UX design is making students learn creating user-friendly and engaging digital products, which will enhance lot of career prospects in product industry , and help them to develop problem-solving and communication skills.		
Module	Name of Module	Lectures
1	Unit - I : Fundamentals of Design Thinking 1.1 Introduction to Design Thinking Concept and Purpose, Five Stages of Design Thinking: Empathize, Define, Ideate, Prototype, Test 1.2 User Requirements and Analysis	15

	Introduction to research and analysis tools (e.g., FigJam) Understanding User Requirements Types of User Research: (a)Qualitative Research (b)Quantitative Research Tools for Collecting User Requirements: (a)Personal Observation (b)Interviews (c)Questionnaires (d)User/Expert Reviews 1.3 Understanding Target Audience and Client Needs Competitive Analysis, Affinity Mapping, Defining User Persona	
2	Unit - II : Introduction to UI 2.1 What is User Interface (UI) Design? Relationship Between UI and UX,Roles in UI/UX 2.2 UI Design Principles Historical Overview of Interface Design Interface Conventions Approaches to Screen-Based UI Template vs Content Formal Elements of Interface Design Active Elements of Interface Design Composing the Elements of Interface Design 2.3 UI Design Process Visual Communication Design in Interface Design	15
3	Unit - III : Introduction to UX & AI in UI Design 3.1 UX Basics Foundation of UX Design Good vs. Poor Design Understanding Users 3.2 Designing the User Experience Elements of User Experience Visual Design Principles Functional Layout Interaction Design 3.3 User Testing and Design Development User Testing Methods Developing and Releasing Designs 3.4 Design Tools & Wireframing Conducting User Interviews Writing Personas: User and Device Personas Understanding User Context Building Low-Fidelity and High-Fidelity Wireframes using Wireframing Tools	15

	3.4 AI Tools in UI/UX Design AI-Powered UI/UX Design Tools: Figma (AI-assisted design features) Adobe XD (AI-driven prototyping) Uizard (AI-generated wireframes) Sketch (AI-powered plugins) Framer AI (AI-driven prototyping and code generation) ChatGPT for UX Writing and Research MidJourney/DALL-E for Generative UI Concepts Creating and Exporting Designs with AI-Powered Prototyping Tools	
Referenc e	Textbook: 1) A Project Guide to UX Design: For user experience designers in the field or in the making (2nd. ed.). Russ Unger and Carolyn Chandler. New Riders Publishing, USA, 2012. 2. The Elements of User Experience: User-Centered Design for the Web and Beyond, Second Edition Jesse James Garrett, Pearson Education. 2011. 3. The Essential Guide to User Interface Design: An Introduction to GUI Design Principles and Techniques, Third Edition Wilbert O. Galitz, Wiley Publishing, 2007. 4. The UX Book Process and Guidelines for Ensuring a Quality User Experience, Rex Hartson and Pardha S. Pyla, Elsevier, 2012	

	Practical of AI based User Interface Design	
Course Code	Course Title	Credits
1	Compare different websites/ Apps under one category for design aesthetics, categories of website/ App such as government / e-commerce / tourism related etc.	
2	Explore any 4 interfaces, find the problems in overall navigation, look and feel of the interface, and relevance of the information. Record all observations using Design tool	
3	Create a 'blank project'. Use frame, shape, text of design tool to create screen layout of given user interface.	
4	Create grid system for the given screen using any design tool	
5	Use frames, images, and colors to design given screen.	
6	Explore various plug-ins/ extensions in the design tool.	
7	Use different plug-ins/extensions in design tool.	
8	Use frames, components, auto-layouts to design given screen using Design tool.	
9	Use interactions, menus to replicate web page design.	
10	Use various menus - bottom menu, slide menu to demonstrate navigations in the screen.	
11	Use components and navigations to design quiz like page in design tool.	
12	Observe gamification techniques used in existing user interfaces.	
13	Use files, templates to create gamification effect in given scenario using design tool.	

14	Use files, templates to create gamification effect in given scenario using design tool.	
15	Observe micro-animations used in existing websites, Apps, interfaces.	
16	Use templates to create micro-animation for given user scenario.	
17	Use Interactions/ events to create Prototype in design tool.	
18	Use plug-in/ extension to convert the created prototype into html page(s).	
19	Use browser to run the generated HTML page(s).	
20	Visit existing interfaces of smart television Record findings related to color scheme, theme, look and feel, location on display of existing interfaces	
21	Prepare user storyboard and user journey mapping for given user interface	
22	Prepare low, medium, and high fidelity prototype for given user interface	
23	Prepare a library/repository of design components using any design tool like Figma.	
24	Reconstruct any ticket booking website to address improvements in look and feel, ease of use within it.	
25	Use Adobe XD's AI features (auto-animate, content-aware layout) to design a dynamic app screen.	
26	Apply Adobe XD's AI-based recommendations for layout, color schemes, and element placements.	
27	Generate microcopy (tooltips, labels, onboarding text) using ChatGPT for a given app interface.	
28	Simulate a user interview or persona creation by prompting ChatGPT for responses. Record insights.	

F. Y. B. Sc. Artificial Intelligence

SEMESTER II

Vocational Skill Development Course

COURSE TITLE: Object-Oriented Programming using JAVA

COURSE CODE: USEAIS2-107

[CREDITS - 02]

Course Learning Objective

The objective of this course is:

To provide fundamental knowledge of Java programming, including object-oriented concepts, GUI development, exception handling, multithreading, and networking. The course aims to develop the ability to design, implement, and manage Java-based applications using modern programming techniques.

Course Learning Outcomes

CO1: Understand core Java concepts including classes, objects, data types, control structures, and OOP principles.

CO2: Apply advanced Java features such as inheritance, interfaces, packages, GUI programming, and event handling.

CO3: Develop Java applications using exception handling, multithreading, networking, and database connectivity (JDBC/ODBC).

Module	Name of Module	Lectures
1	1. Basic Java Programming 1.1 JAVA Features and JAVA Programming Environment 1.2 Defining a class, creating an object, accessing class member 1.3 Java Tokens and data types 1.4 Operators and expressions 1.5 Decision making and looping	10
2	2. Inheritance, Interface, Packages, AWT, Swing and Event Listeners 2.1 Inheritance: the concept of inheritance, types of inheritance 2.2 Interfaces 2.3 Package 2.4 Component, container 2.5 Introduction to Swing, Swing AWT 2.6 The delegation Event Model: Event sources, Event list	10

	2.7 Event classes 2.8 Event listener interfaces	
3	Exception Handling, Multithreading and Networking 3.1 Errors and exception 3.2 Multithreaded programming 3.3 Socket Overview: Datagram: DatagramPacket, Datagram Server and Client. 3.4 Introduction to JDBC and ODBC	10
Reference	Textbook: Textbooks: 1. Java Programming, Dr. RAjendra Kawale, Devra} 2. Programming with Java, Balgurusamy, Tata Mc-Graw Hill 3. Java2 Programming BlackBook, Holzner, Stavenetal, Dreamtech Press New Delhi	
Reference	Website: 1. www.w3schools.com 2. www.javatpoint.com 3. www.geeksforgeeks.org	
Course Code	Course Title	Credits
USEAIS2-107	Practical of Object Oriented Programming using JAVA	1
1	Setup Java Programming development environment using: ● Command prompt. (Classpath and path setup) ● Any IDE (Eclipse, Netbeans, VScode, Jcreator, etc.).	
2	Write programs to demonstrate the use of: If statements (all forms of if statement) ● Switch – Case statement ● Different types of Loops(for, while, and do..while).	
3	Write programs for implementation of different methods of: ● String class. ● StringBuffer class.	
4	Write programs to demonstrate: ● Use of Array. ● Use of Vectors.	

5	Write programs using Wrapper Class : ● To convert a primitive into an object. ● To convert objects into primitive.	
6	Develop a program to implement: ● Single inheritance. ● Multilevel inheritance.	
7	Develop a program for the implementation of the interface.	
8	Write programs to demonstrate the use of : ● Built-in packages ● User-defined packages.	
9	Write programs for implementation of try, catch, and finally block.	
10	Write programs for implementation of throw, throws clause	
11	Write programs using multithreading.	
12	Write a program to design any type of form using AWT components.	
13	Write a program to demonstrate the use of border layout. The layout shows four buttons at four sides with captions “left”, “right”, “top” and “bottom” using Swing Components.	
14	Write a program to design a calculator to demonstrate the use of grid layout using swing components.	
15	Write a program to handle key events and mouse events.	
16	Write a program to implement action events in the frame using swing components.	
17	Write a program to retrieve the hostname and IP address using the InetAddress class.	
18	Write a program to demonstrate various methods of: ● URL class. ● URLConnection.	
19	Write a program that demonstrates connection-oriented communication using a socket.	
20	Write a program to demonstrate sending and receiving data through Datagram.	
21	Write a program to: ● Create a sample database. ● Make connectivity with the database.	

22	Write a program to implement the following operations on the database: ● Insert record. ● Update record. ● Delete record.	
23	Write a program to demonstrate the use of PreparedStatement.	
24	Write a program to retrieve data from a table using the ResultSet interface. (Use various methods of navigation methods).	