



।न हि ज्ञानेन सदृशं । पवित्रमिह विद्यते।

Shriram Shikshan Sanstha's



Shriram Institute of Information Technology, Paniv

Tal-Malshiras Dist-Solapur [MH] 413113

NAAC Accredited with 'B' Grade

2.6.1 - Programme & course outcomes BCA (Bachelor in Computer Application)

Program Outcomes: -

- This is a three years bachelor degree course in computer applications aimed at developing computer professional versatile in use of computers mostly in business world.
- The emphasis is to have generality of developing professionals as programmer, system analysts, database administrators, documentation officer, etc.

Program specific outcomes: -

- To pursue further studies to get specialization in Computer Science, Computer Applications, Personal Management, Tally, Business Administration
- To work in IT sector as a Software Developer, Software Tester, Software Analyzer, Junior programmer, Web developer, system administrator.
- To work in public as well as private sector as a computer operator and also in banking sectors.

Bachelor in Computer Application B.C.A – I (Science) Sem- I to Sem - VI

Sr. No	Course Name	Sem	Course Outcomes
1	Fundamentals of Computer	I and II	<ul style="list-style-type: none">• To understand basic concepts and terminology of information technology.• To a basic understanding of personal computers and their operations.• To understand various input and output devices.• To understand internet concepts.
2	Office Automation		<ul style="list-style-type: none">• Integrate both graphs and tables created in Microsoft Excel into a laboratory report in Microsoft Word.• Generate equations, sample calculations, and basic diagrams in Microsoft Word.• Input experimental data into Microsoft Excel.• Perform calculations in Microsoft Excel using both manually



		<ul style="list-style-type: none"> inputting formulas and built- in Functions. • Generate simple and effective tables and graphs to describe experimental data in Microsoft Excel. • Properly format and organize a formal laboratory report in Microsoft Word.
3	Programming and Problem Solving using 'C'-I	<ul style="list-style-type: none"> • Able to understand the basic concepts of C programming language. • Enhance skill on problem solving by constructing algorithms • Students will be able to comprehend the general structure of C program, concepts of variable, datatype, operator and be able to create a C program to demonstrates these concepts. • Able to design and develop various programming problems using C programming concepts. • Understand and use various constructs of the programming language such as conditionals, iteration. • Demonstrate the use of strings and string handling functions • Apply skill of identifying appropriate programming constructs for problem solving
4	Web Programming-I	<ul style="list-style-type: none"> • Analyze a web page and identify its elements and attributes. • Create web pages using HTML and Cascading Style Sheets. • Build static web pages using HTML,CSS (Client side programming). • Create XML documents and Schemas.
5	Basics of Mathematics	<ul style="list-style-type: none"> • To provide overview of theory of discrete objects, starting with relations and partially ordered sets. • To describe the fundamental counting principle and to determine the number of possible combinations for a given situation using the fundamental counting principle • Understand the basic principles of sets and operations in sets. • Prove basic set equalities. • Demonstrate an understanding of relations able to determine their properties.
6	Descriptive Statistics	<ul style="list-style-type: none"> • To prepare frequency distribution and represent it by graphically with the help of tables. • To compute various measures of central tendency, dispersion and to interpret them. • To compute correlation coefficient and interpret its value. • To estimate or predict through linear regression method.
7	Fundamentals of Electronics	<ul style="list-style-type: none"> • Learn how to develop the Integrated circuits (IC) in electronics



		<p>systems. E.g. Computer system, Microprocessor, Microcontroller, Mobile etc.</p> <ul style="list-style-type: none"> • Learn how to Manufacturing Resistors, Capacitors, Diode and Transistor in IC. • An understanding of different Display devices, Sensors and PCB technologies used In Computer System.
8	Linear Electronics	<ul style="list-style-type: none"> • Learn how to develop and employ circuit models for elementary electronic Components, e.g., resistors, inductors, capacitors, diodes and transistors. • Gain an intuitive understanding of the role of power flow and energy storage In electronic circuits. • Learn how to develop different power supplies in computer system.
9	Introduction to Python Programming	<ul style="list-style-type: none"> • Understand the features or characteristics of Python. • Understand the concept of Python Virtual Machine, Python Data types, Command Line Argument, Operators. • Explore Integrated Development Environment (IDE). • Do programs using conditional control statements and also use the concept of Looping for doing programs. • Describe the concept of strings, Collection Lists, Tuples and Dictionaries.
10	Operating System	<ul style="list-style-type: none"> • Understand fundamental operating system abstractions such as processes, threads, files, semaphores, IPC abstractions, shared memory regions, etc. • To provide a sound understanding of the Computer operating system, its structures, and its functioning. • Analyze Process scheduling algorithms. • To understand what a process is and how processes are synchronized and scheduled.
11	Programming and Problem Solving using 'C'-II	<ol style="list-style-type: none"> 1. Able to Implement advance C programming concepts like function, pointer, structure and union etc. 2. Understand the dynamics of memory by the use of pointers. 3. Able to understand the file handling using C Programming language. 4. To understand the concept of macros and preprocessor.
12	Web Programming-II	<ul style="list-style-type: none"> • Develop programming skills by the use of java script • Build dynamic web pages using JavaScript (Client side



			<p>programming).</p> <ul style="list-style-type: none"> Analyze to Use appropriate client-side applications. Build interactive web applications using JQuery. Develop solution to complex problems using appropriate method, technologies, frameworks, web services and content management Extend this knowledge to .Net Platforms, Java Technologies, Full Stack Development
13	Graph Theory		<ul style="list-style-type: none"> Understand the notion of mathematical thinking, mathematical proofs and to apply them in problem solving. Ability to understand and apply concepts of graph theory in solving real world problems ability to reason logically. Apply the concepts of graph theory in data structure of computer science. Give an understanding of graphs and trees which are widely use in software.
14	Probability Theory		<ul style="list-style-type: none"> To distinguish between random and non-random experiments. To find the probabilities of the events. To apply discrete and continuous probability distributions studied in this course in different situations.
15	Digital Fundamentals of Computer		<ul style="list-style-type: none"> Design and constructs logic as well as arithmetical circuits Calculate various important parameters of Digital logic families Design & analyze combinational logic circuits Design & analyze sequential logic circuits To Executed 8085 Microprocessor Assembly language programming.
16	Introduction to Microprocessor and Interfacing		<ul style="list-style-type: none"> Design, test and critically evaluate embedded solutions to real world situations using digital components (sequential and combinational). Recognize the key features of embedded systems in terms of computer hardware and be able to discuss their functions. You will be aware of the key factors affecting computing hardware evolution. Design, test and critically evaluate embedded solutions to real world situations using (embedded) computer systems interfaced to digital hardware
17	OOPS with C++-I and II	III and IV	<ul style="list-style-type: none"> Describe the procedural and object oriented paradigm with concepts of streams, classes, functions, data and objects. Understand Dynamic memory management techniques using pointer, constructor and destructor.



		<ul style="list-style-type: none"> • Describe the concept of function overloading, operator overloading, virtual function and polymorphism. • Classify inheritance with the understanding of early and late binding, uses of exception handling. <p>Demonstrate the use of OOPs concepts with the help of programs.</p>
18	Data structures using 'C'- I and II	<ul style="list-style-type: none"> • Understand the concept of Data, types of Data, ADT, Algorithm. • Understand the concept of Complexity. • Understand basic data structures such as array, stack, queue and linked list. • Solve the problems involving graphs and trees. <p>Apply Algorithms for solving problems like sorting, searching, insertion and deletion of data.</p>
19	Database Management System	<ul style="list-style-type: none"> • Understand the concept of Data, Database, and Database Management System. • Understand the concept of Database structures, Database architecture – 2 and 3 – tier, Database design, Relational models. • Understand the concept of transaction, ACID concept and transaction states. • Understand the concept of concurrency control, Deadlock, Deadlock handling methods. • Understand the concept of Database recovery and the concept of Distributed Databases.
20	MySQL	<ul style="list-style-type: none"> • Understand how MySQL is installed. • Implement basic DDL, DML, DCL and DQL commands. • Understand how data is inserted, updated, deleted to or from the database and also apply constraints on columns. • Write sub –queries and understand their purpose. • Understand the concept of MySQL Join and View that is how create, update and alter and also drop it. • Understand the concept of Union, Indexing. • Doing implementation on database using stored procedure, How to import and export database from MySQL.
21	Software Testing & Quality Assurance	<ul style="list-style-type: none"> • Describe fundamental concepts of software quality assurance. • Explore test planning and its management. • Understand fundamental concepts of software automation. • Apply Selenium automation tool for testing web based application.



		<ul style="list-style-type: none"> • Demonstrate the quality management, assurance, and quality standard to software system. • Demonstrate Software Quality Tools and analyze their effectiveness.
18	Web Development using PHP	<ul style="list-style-type: none"> • Understand the concept of Web application, WebServers, IIS and PWS. • Understand the concept of Data types in PHP, Control statements, Array and functions. • Understand the concept of String, working with forms and Validation concept. • Understand the concept of combine HTML and PHP code. • Understand the concept of History of MYSQL, Architecture, Connecting to MYSQL. • Understand the concept of state management, cookies, session.
19	Computer Networks-I and Advanced Computer Networks	<ul style="list-style-type: none"> • Build an understanding of the fundamental concepts of computer networking. • Familiarize the student with the basic taxonomy and terminology of the computer networking area. • Introduce the student to advanced networking concepts, preparing the student for entry Advanced courses in computer networking. • Independently understand basic computer network technology. • Identify the different types of network topologies and protocols. • Enumerate the layers of the OSI model and TCP/IP. Explain the function(s) of each layer.
20	Financial Accounting with Tally	<ul style="list-style-type: none"> • Understand the concept of Book-keeping, Accountancy, AS, IFRS. • Identify and understand of double entry Book – keeping system and their methods. • Understand the concept of Journal, Ledger, Subsidiary – Books, Bank Reconciliation statement. • Understand the concept of effect of error , types. • Understand the concept of final accounts of proprietary concern, TDS. • Doing implementation through Tally like create, alter and display stock items, etc.
21	Ethics and Cyber law	<ul style="list-style-type: none"> • Students identify and analyze statutory, regulatory, constitutional, and organizational laws that affect • the information technology professional. • Students locate and apply case law and common law to current legal dilemmas in the technology



		<ul style="list-style-type: none"> • field. • Students apply diverse viewpoints to ethical dilemmas in the information technology field and recommend appropriate actions. • Students distinguish enforceable contracts from non-enforceable contracts. • Students demonstrate leadership and teamwork.
22	Angular JS	<ul style="list-style-type: none"> • Reduce the amount of code you write to build rich user interface applications. • Increase the reliability and maintainability of UI by using data binding. • Retrieve data from back-end server, manipulate it and display it with ease. • Modularise your code with the custom services and directives. • Providing two way binding of data. • Create Single Page Applications (SPA).
23	Environmental Studies	<ul style="list-style-type: none"> • Conceptualize the process and various factors involved in the formation of environment. • Recognize the importance of environment and the sustainable of natural resources. • Analyze interaction between social and environmental process. • Use scientific reasoning to identify and understand environment problems and evaluate potential solutions. • Visualize the impact of human activities on environment and role of society in these impact. • Recall critically about their role as citizens, consumers and environmental actors in a and inter connected world.
24	Python Programming	<ul style="list-style-type: none"> • Understand the features or characteristics of Python. • Observe the process of How to install python and how it works. • Understand the concept of Python Virtual Machine, Python Data types, Command Line Argument, Operators. • Do programs using conditional control statements and also use the concept of Looping for doing programs. • Describe the concept of strings, Collection Lists, Tuples and Dictionaries. • Describe how function is used in Python programming, how it is implemented, Modules.



			<ul style="list-style-type: none"> • Describe the concept of OOP in Python like class, object, constructors, inheritance concept. • Implement file operations using Python. • Describe the concept of Exception Handling, Types of Exception handling, How exception is handled and overloading concepts.
25	Core Java and Advanced Java	V and VI	<ul style="list-style-type: none"> • Understand the features of Java and the architecture of JVM • Write, compile, and execute Java programs that may include basic data types and control flow constructs and how type casting is done. • Identify classes, objects, members of a class and relationships among them needed for a specific problem and demonstrate the concepts of polymorphism and inheritance. • The students will be able to demonstrate programs based on interfaces and threads. • Write, compile, execute Java programs that include GUIs and event driven programming and also programs based on files • Learn to access database through Java programs, using Java Data Base Connectivity (JDBC). • Create dynamic web pages, using Servlets and JSP. Make a reusable software component, using Java Bean. • Develop Stateful, Stateless and Entity Beans. Use Struts frameworks, which give the opportunity to reuse the codes for quick development. Map Java classes and object associations to relational database tables with Hibernate mapping files.
26	Visual Programming		<ul style="list-style-type: none"> • Understand code solutions and compile C# projects within the .NET framework. • Design and develop professional console and window based .NET application • Demonstrate knowledge of object-oriented concepts Design user experience and functional requirements C#.NET application. • Construct classes, methods, and assessors, and instantiate objects. • Understand and implement string manipulation, events and exception handling within .NET application environment. • Create and manipulate GUI components in C#. • Design and Implement Windows Applications using Windows Forms, Control Library, Advanced UI Programming & Data Binding concepts. • Design and Implement database connectivity using ADO.NET in window based application.



27	Computer Graphics	<ul style="list-style-type: none"> • The main objective of this module is to introduce to the students the concepts of computer graphics. • Various applications, areas, and graphics displays and hardcopy technologies. • Demonstrate the overview of graphics system and make use of various drawing algorithms of output primitives • Apply and compare the algorithms for drawing 2D images also explain aliasing, anti-aliasing and half toning techniques. • Experiment with the geometric transformations and different algorithms for viewing and clipping in two dimensional graphics related problems. • Solve the problems on viewing transformations and explain the projection and hidden surface removal algorithms.
28	Recent Trends in IT	<ul style="list-style-type: none"> • It begins by providing an overview of recent trends in the access and use of new technologies as well as a summary of online opportunities and risks. • Pursue advanced knowledge and professional development in the field of information technology. • It then explores a variety of factors, including economic, social and cultural status which underlie these trends and lead to online and offline inequalities. • Create, select and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations. • Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
29	Linux & Shell Programming	<ul style="list-style-type: none"> • To provide introduction to Linux Operating System and its File System. • To develop the ability to formulate Regular Expressions and use them for Pattern Matching • Understand the Linux commands • To acquire skills shell programming • To implement various system calls. • To learn basics concepts of system administration, networking
30	Dot Net Technology	<ul style="list-style-type: none"> • Understand web concepts and features of ASP.NET • Implement web applications using various ASP.NET controls • Implement web applications using ASP.NET MVC



		<ul style="list-style-type: none">• Set up various navigation techniques for integrating web pages within the site.• Create the dynamic web page using ASP.NET Controls which interact with databases.• Manage cookies and sessions as state management techniques.• Advanced concepts related to Web Services, WCF and WPF in project development
31	Data Warehouse and Data Mining	<ul style="list-style-type: none">• Identify the scope and necessity of Data Mining & Warehousing for the society• To understand various tools of Data Mining and their techniques to solve the real time problems.• Pre-process the data for mining applications.• Apply the association rules for mining the data.• Design and deploy appropriate classification techniques.• Cluster the high dimensional data for better organization of the data.• Evaluate various mining techniques on complex data objects
32	Cryptography and Network Security	<ul style="list-style-type: none">• Illustrate the principles of number theory and compare various cryptographic techniques.• Demonstrate how Block Ciphers such as DES, AES, Triple DES, RC5 and public keycrypto-systems are implemented.• Apply hash function and digital signatures to implement authentication protocols• Illustrate the role of firewall in implementing trusted systems• Analyze how applications can be secured
33	Advanced Python	<ul style="list-style-type: none">• Students should be made familiar with the concepts of GUI controls and designing GUI applications.• To develop the web applications• Understanding the XML Concepts• To understand and implement the concept of networking in Python.



B.Sc(E.C.S) – B.Sc(Entire Computer Science)

Program Outcomes: -

- In this course students get the knowledge about various courses that is Data Structure using C++, Software Engineering and Testing, Database Management System, MySQL, Operating System, Linux OS and Shell Scripting, Web Development using PHP, etc that are useful for taking a job in IT sectors.
- Students who interested for doing job in IT sectors that students can do job.
- Students also work as a software engineer, software tester, web developer, software developer, etc.
- Students also work in public as well as private sector as a computer operator.

Program specific outcomes: -

Ability to apply the knowledge gained during the course of the program from Data Structures using C++, Software Engineering and Testing, Probability theory and Introduction to Python Programming, DBMS, MySQL, Operating System, Statistics for Data Science, Optimization techniques, PHP, Environmental studies, etc

Sr. No	Course Name	Sem	Course Outcomes
1	Fundamental of Computer	I and II	<ul style="list-style-type: none"> • To understand basic concepts and terminology of information technology. • To a basic understanding of personal computers and their operations. • To understand various input and output devices. • To understand memory management.
2	Basics of Operating System		<ul style="list-style-type: none"> • To provide a sound understanding of the Computer operating system, its structures, and its functioning. • To understand the services provided by and the design of an operating system. • To understand different approaches to memory management. • To understand the services provided by and the design of



		<p>an operating system.</p> <ul style="list-style-type: none"> To understand what a process is and how processes are synchronized and scheduled.
3	Programming using 'C'	<ul style="list-style-type: none"> To understand the fundamentals of C programming. To read, Understand, Write and Execute the programs using C. To apply logical thinking to a given program and write the code. To Identify the correct and efficient ways of solving problems.
4	Python - I	<ul style="list-style-type: none"> Understand the basic concepts and applications of Python. Design, create, build, and debug python applications. Explore Integrated Development Environment (IDE). Write and apply decision structures for different operations. Write loop structures to perform iterative tasks.
5	Numerical Methods	<ul style="list-style-type: none"> Ability to appreciate real world applications which use these concepts. Skill to formulate a problem through Mathematical Modeling and programming.
6	Graph Theory	<ul style="list-style-type: none"> Understand the notion of mathematical thinking, and mathematical proofs and to apply them in problem solving. Ability to understand and apply concepts of graph theory in solving real world problems and ability to reason logically.
7	Basic Electronics	<ul style="list-style-type: none"> Learn how to develop and employ circuit models for elementary electronic components, e.g., resistors, inductors, capacitors, diodes and transistors. Gain an intuitive understanding of the role of power flow and energy storage in electronic circuits. Learn how to develop different power supplies in the computer system.



8	Advanced Electronics		<ul style="list-style-type: none"> • Learn how to develop the Integrated circuits (IC) in electronics systems. E.g. Computer systems, Microprocessor, Microcontroller, Mobile, etc. • Learn how to Manufacturing Resistors, Capacitors, Diode and Transistor in IC. • An understanding of different Display devices, Sensors and PCB technologies used in Computer System.
	Introduction to Web Technology		<ul style="list-style-type: none"> • Explain the history of the internet and related internet concepts that are vital in understanding web development. • Discuss the insights of internet programming and implement complete applications over the web. • Demonstrate the important HTML tags for designing static pages and separate design from content using Cascading Style sheet. • Utilize the concepts of JavaScript.
	Operating System		<ul style="list-style-type: none"> • Describe and analyze memory management and its allocation policies. • Identify the use and evaluate the storage management policies concerning different storage management technologies. • To understand different approaches to memory management.
	Object Oriented Programming using C++		<ul style="list-style-type: none"> • Describe the procedural and object oriented paradigm with concepts of streams classes, functions, data and objects. • Understand dynamic memory management techniques using pointers, constructors, destructors, etc. • Describe the concept of function overloading, operator overloading, virtual functions and polymorphism. • Classify inheritance with the understanding of early and late binding, usage of exception handling, and generic programming. • Demonstrate the use of various OOPs concepts with the help of programs
	Python - II		<ul style="list-style-type: none"> • Write and implement a functional approach to application development.





			<ul style="list-style-type: none"> To Execute 8085 Microprocessor Assembly language programming.
9	Data Structure using C++ -I and II		<ul style="list-style-type: none"> Understand the concept of Data, types of Data, ADT, Algorithm. Understand the concept of Complexity. Understand basic data structures such as array, stack, queue and linked list. Solve the problems involving graphs and trees. Apply Algorithms for solving problems like sorting, searching, insertion and deletion of data.
10	Software Engineering	III and IV	<ul style="list-style-type: none"> Understand the concept of System, Elements, types of System. Describe and understand the concept of System analysis and role of system analyst. Plan a Software engineering process life cycle, including the specification, design, implementation, and testing of software systems that meet specification, performance, maintenance and quality requirement. Draw a decision table and decision tree from the information. Arrange the information or data using normalization that is in 1NF, 2NF, 3NF, 4NF and 5NF. Understand the concept of coding, construction of system, SQA.
11	Software Testing		<ul style="list-style-type: none"> Understand the Concept of testing, importance of testing, concept of WBT, BBT and levels of testing. Understand the concept of Functional and Non-functional testing. Understand the concept of Test cases, test case template. Understand the concept of Bug, Defect, Error and fault.
12	Probability Theory -I and II		<ul style="list-style-type: none"> Understand and solve the examples using Permutations and Combinations. Describe the concept of Sample space, types, events, types of events. Understand the concept of Conditional probability, Independence events. Study the various special probability distributions such as Binomial, Poisson, Hypergeometric



		<ul style="list-style-type: none"> • Write and implement a modular approach to application development. • Design an application using object-oriented paradigm. • Create error free applications by applying the exception handling concept. • Design an application that contains the use of different files for data processing.
	Linear Algebra	<ul style="list-style-type: none"> • Understand the notion of mathematical thinking, mathematical proofs, and algorithmic thinking, and be able to apply them in problem solving. • Understand the basics of combinatorics, and be able to apply the methods from these subjects in problem solving. • Be able to use effectively algebraic techniques to analyse basic discrete structures and algorithms. • Understand asymptotic notation, and its significance, and be able to use it to analyse asymptotic performance for some basic algorithmic examples. • Understand some basic properties of graphs and related discrete structures, and be able to relate these to practical examples.
	Discrete Mathematics	<ul style="list-style-type: none"> • To understand the notion of mathematical thinking, mathematical proofs, and algorithmic thinking, and be able to apply them in problem solving. • To understand the basics of combinatorics, and be able to apply the methods from these subjects in problem solving. • To use effectively algebraic techniques to analyse basic discrete structures and algorithms. • To understand asymptotic notation, and its significance, and be able to use it to analyse asymptotic performance for some basic algorithmic examples. • To understand some basic properties of graphs and related discrete structures, and be able to relate these to practical examples.
	Digital Electronics and Microprocessor	<ul style="list-style-type: none"> • Design and construct logic as well as arithmetical circuits • Calculate various important parameters of Digital logic families • Design & analyze combinational logic circuits • Design & analyze sequential logic circuits



			distributions.
13	Introduction to Python Programming		<ul style="list-style-type: none"> • Understand and solve numerical problems. • Introduction to Python programming. • Understand the features or characteristics of Python. • Observe the process of How to install python and how it works. • Understand the concept of Python Virtual Machine, • Python Data types, Command Line Argument, Operators. • Do programs using conditional control statements and also use the concept of Looping for doing programs. • Describe the concept of strings, Collection Lists, Tuples and Dictionaries. • Describe how function is used in Python programming, how it is implemented, Modules. • Describe the concept of OOP in Python like class, object, constructors, inheritance concept. • Implement file operations using Python. • Describe the concept of Exception Handling, Types of Exception handling, How exception is handled and overloading concepts.
14	Database Management System		<ul style="list-style-type: none"> • Understand the concept of Data, Database, Database Management System. • Understand the concept of Database structures, • Database architecture – 2 and 3 – tier, Database design, Relational models. • Understand the concept of transaction, ACID concept and transaction states. • Understand the concept of concurrency control, Deadlock, Deadlock handling methods. • Understand the concept of Database recovery and the concept of Distributed Databases.
15	MySQL		<ul style="list-style-type: none"> • Understand how MySQL is installed. • Implement basic DDL, DML, DCL and DQL commands. • Understand how data is inserted, updated, deleted to or from the database and also apply constraints on columns. • Write sub –queries and understand their purpose. • Understand the concept of MySQL Join and View that is how create, update and alter and also drop it. • Understand the concept of Union, Indexing.



		<ul style="list-style-type: none"> • Doing implementation on database using stored procedure, How to import and export database from MySQL.
16	Operating System	<ul style="list-style-type: none"> • Understand basics of Operating system, types, System calls. • Understand the concept of Process and thread management. • Explain the concept of Process synchronization, Deadlock, Deadlock prevention and detection methods. • Explain various memory management techniques and concept of virtual memory and Thrashing. • Use of disk management and disk scheduling algorithms for better utilization of external memory.
17	Linux OS and Shell Scripting	<ul style="list-style-type: none"> • To provide introduction to Linux Operating System and its File System. • To develop the ability to formulate Regular Expressions and use them for Pattern Matching • Understand the Linux commands • To acquire skills shell programming • To implement various system calls. • To learn basics concepts of system administration, networking.
19	Optimization techniques	<ul style="list-style-type: none"> • Explain the concept of O. R with history, evolution, scope and limitations. • Understand the concept of LPP, Graphical methods – Simplex, Big – M method. • Understand the concept of TP, MODI, Vogels approximation method and solve examples. • Understand the concept of AP and solve examples using AP.
20	Web Development using PHP	<ul style="list-style-type: none"> • Understand the concept of Web application, WebServers, IIS and PWS. • Understand the concept of Data types in PHP, Control statements, Array and functions. • Understand the concept of String, working with forms and Validation concept.



			<ul style="list-style-type: none"> • Understand the concept of combine HTML and PHP code. • Understand the concept of History of MYSQL, Architecture, Connecting to MYSQL. • Understand the concept of state management, cookies, session.
21	Environmental Studies		<ul style="list-style-type: none"> • Conceptualize the process and various factors involved in the formation of environment. • Recognize the importance of environment and the sustainable of natural resources. • Analyze interaction between social and environmental process. • Use scientific reasoning to identify and understand environment problems and evaluate potential solutions. • Visualize the impact of human activities on environment and role of society in these impact. • Recall critically about their role as citizens, consumers and environmental actors in a and inter connected world.
22	Data Communication and Networking	V and VI	<ul style="list-style-type: none"> • Familiarize with contemporary issues in network technologies. • Know the layered model approach explained in OSI and TCP/IP network models • Identify different types of network devices and their functions within a network. • Know the Basic routing mechanisms, IP addressing scheme and internetworking concepts. • Familiarize with IP and TCP Internet protocols. • Understand major concepts involved in design of WAN, LAN and wireless networks. • Know the basics of network configuration and maintenance. • Know the fundamentals of network security issues.
23	Theory of Computer Science		<ul style="list-style-type: none"> • Understand the basic concepts and application in Theory of Computation. • Apply this basic knowledge of Theory of Computation in the computer field to solve computational problems and in the field of compiler also.
24	Visual Programming		



		<ul style="list-style-type: none">• Design, create, build, and debug Visual programming applications.• Explore Visual Programming Integrated Development Environment (IDE).• Implement syntax rules in Visual programming.• Explain variables and data types used in program development.• Write and apply decision structures for determining different operations.• Write and apply loop structures to perform repetitive tasks.• Write and apply procedures, sub-procedures, and functions to create manageable code.
25	Advanced Java	<ul style="list-style-type: none">• Design, create, build, and debug Java applications.• Explore Integrated Development Environment (IDE).
26	Advanced Python Programming	<ul style="list-style-type: none">• Design, Create, Build, and Debug Python applications.• Explore Integrated Development Environment (IDE).
27	System Security	<ul style="list-style-type: none">• Develop an understanding of information assurance as practiced in computer operating systems, distributed systems, networks and representative applications.• Gain familiarity with prevalent network and distributed system attacks, defenses against them, and forensics to investigate the aftermath.• Develop a basic understanding of cryptography, how it has evolved, and some key encryption techniques used today.• Develop an understanding of security policies (such as authentication, integrity and confidentiality), as well as protocols to implement such policies in the form of message exchanges.
28	Compiler Construction	<ul style="list-style-type: none">• To gives you with both theoretical and practical knowledge that is crucial in order to implement a



		<ul style="list-style-type: none">• Design, create, build, and debug Visual programming applications.• Explore Visual Programming Integrated Development Environment (IDE).• Implement syntax rules in Visual programming.• Explain variables and data types used in program development.• Write and apply decision structures for determining different operations.• Write and apply loop structures to perform repetitive tasks.• Write and apply procedures, sub-procedures, and functions to create manageable code.
25	Advanced Java	<ul style="list-style-type: none">• Design, create, build, and debug Java applications.• Explore Integrated Development Environment (IDE).
26	Advanced Python Programming	<ul style="list-style-type: none">• Design, Create, Build, and Debug Python applications.• Explore Integrated Development Environment (IDE).
27	System Security	<ul style="list-style-type: none">• Develop an understanding of information assurance as practiced in computer operating systems, distributed systems, networks and representative applications.• Gain familiarity with prevalent network and distributed system attacks, defenses against them, and forensics to investigate the aftermath.• Develop a basic understanding of cryptography, how it has evolved, and some key encryption techniques used today.• Develop an understanding of security policies (such as authentication, integrity and confidentiality), as well as protocols to implement such policies in the form of message exchanges.
28	Compiler Construction	<ul style="list-style-type: none">• To gives you with both theoretical and practical knowledge that is crucial in order to implement a

			programming language. <ul style="list-style-type: none"> It gives you a new level of understanding of a language in order to make better use of the language (optimization is just one example).
29	Internet Programming using ASP.Net		<ul style="list-style-type: none"> Create Design, Debug and Deploy Web applications. Explore Integrated Development Environment (IDE).
30	Angular JS		<ul style="list-style-type: none"> Create Design, Debug and Deploy Web applications. Explore Integrated Development Environment (IDE).
31	Mobile Application Development		<ul style="list-style-type: none"> Create, Design, Debug and Deploy Android applications. Explore Integrated Development Environment (IDE).



M.Sc (Computer Science)

Program Outcomes: -

- Design and develop computer based system in the areas related to OOP using C++.
- Students can gain knowledge about new technology and also develop their skills.

Program specific outcomes: -

Ability to apply the knowledge gained during the course of the program from Object Oriented Programming using C++, Advanced DBMS, Data Structures and Algorithms, Software Engineering, Java Programming, Python Programming, Computer Communication networks, AI, OA, etc.

Sr. No	Course Name	Sem	Course Outcomes
1	Object Oriented Programming using C++	I and	<ul style="list-style-type: none"> Describe the procedural and object oriented



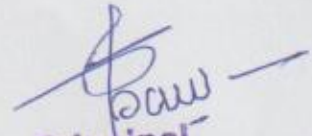
		II	<p>paradigm with concepts of streams, classes, functions, data and objects.</p> <ul style="list-style-type: none"> • Understand Dynamic memory management techniques using pointer, constructor and destructor. • Describe the concept of function overloading, operator overloading, virtual function and polymorphism. • Classify inheritance with the understanding of early and late binding, uses of exception handling. • Demonstrate the use of OOPs concepts with the help of programs.
2	Advanced DBMS		<ul style="list-style-type: none"> • Understand the concept of Data, Database, Database Management System. • Understand the concept of Database structures, Database design, Relational models. • Understand the concept of transaction, ACID concept and transaction states. • Understand the concept of concurrency control, Deadlock, Deadlock handling methods. • Understand the concept of Database recovery and the concept of Distributed Databases.
3	Data Structures and Algorithms		<ul style="list-style-type: none"> • Understand the concept of fundamental notations such as array, stacks. • Understand basic data structures such as array, stack, queue and linked list. • Solve the problems involving graphs and trees. • Apply Algorithms for solving problems like sorting, searching, insertion and deletion of data.
4	Software Engineering		<ul style="list-style-type: none"> • Plan a Software engineering process life cycle, including the specification, design, implementation, and testing of software systems that meet specification, performance, maintenance and quality requirement. • Able to collect, analyze and specify software requirements through a productive working relationship with various stakeholders of the project. • Analyze and translate a specification into a design, and then realize that design practically, using as appropriate software engineering methodology. • Know how to develop the code from the design and effectively apply relevant standards and perform testing, and quality management and practice.



		<ul style="list-style-type: none"> • Understand the concept of Software testing, methods such as WBT, Basis path testing, Control testing, BBT, etc.
5	UML	<ul style="list-style-type: none"> • Understand the concept of Object Oriented Fundamentals, Classes, OO modeling. • Describe the concept of Building block of UML. • Understand the architecture of UML and Software development life cycle. • Understand the concepts of basics structural modeling and advanced structural modeling. • Explain how to use case diagrams. • Explain the concepts of events, signals, state machines, time, space and architectural modelings.
6	Java Programming	<ul style="list-style-type: none"> • Understand the concept of Features of Java, Keywords, operators, decision making and looping. • Explain the concept of Inheritance with their types. • Understand the concept of exception handling in Java, how to handle event. • Understand the concept of multithreading, synchronization, Input / Output. • Explain the concept of JDBC, API, JDBC drivers, creating and executing SQL statements.
7	Python Programming	<ul style="list-style-type: none"> • Understand the concept of String, List, Dictionary, Function, Module, Set, Package. • How to handle exception using Python programming using different functions. • Explain the concept of regular expression. • Explain the OOP concept in python such as Class, Object, Inheritance, Method overloading, Constructors, Abstract class, etc. • How to use generators and threads in python programming. • Explain the concept of Data science using Python and NumPY and Django concept in details.
8	Computer Communication Networks	<ul style="list-style-type: none"> • Understand the concept of Computer network basics, Types of Network. • Explain the Data Link layer with design issues, protocols of data link layer.



		<ul style="list-style-type: none">• Understand how network layer is implemented, routing algorithms, Internetworking, etc.• Understand the concept of Transport layer, services, protocols – TCP in details.• Understand the concept of E- mail, WWW, static and web documents, HTTP, Wireless web.
9	Artificial Intelligence	<ul style="list-style-type: none">• Understand the concept of AI, techniques, problems spaces and search.• Explain static search techniques.• Explain statistical reasoning using various theorms.• Understand the concept of natural languages and expert systems.
10	Office Automation	<ul style="list-style-type: none">• Understand the concept of Computer concept in details.• Understand the concept of Working with documents, how to save, how to work using Microsoft word.• Understand the concept of Mail merge.• Explain the basic concept of Microsoft excel, How formulas are implemented, how to handle sheet.• Explain basics points of Powerpoint presentation, how to create a slide, presentation, how to give animation to presentation, how insert pictures, tables, clipArts in the presentation.• Understand the concept of how to import data from other databases.


Principal

Shriram Institute of Information Technology,
Paniy, Tal. Malshiras, Dist. Solapur