

**SEMESTER - IV**

<b>Branch: BCA</b>	<b>Semester-IV</b>
<b>Subject Code: 4101</b>	<b>Lecture: 04 Credit: 04</b>
<b>Course Opted</b>	<b>Core Course - 11</b>
<b>Subject Title</b>	<b>PYTHON PROGRAMMING</b>

**Course Objective:**

- Learn the fundamentals of writing Python scripts.
- Learn core Python scripting elements such as variables and flow control structures.
- Discover how to work with lists and sequence data.
- Write Python functions to facilitate code reuse.

**Course Outcome:**

- Able to apply the principles of pythonprogramming.
- Write clear and effective pythoncode.
- Create applications using pythonprogramming.
- Implementing database usingSQLite.
- Access database using pythonprogramming.
- Develop web applications using pythonprogramming.
- Develop and use Web Services usingpython.

<b>Modules</b>	<b>Sr. No.</b>	<b>Topic and Details</b>	<b>No. of Lectures Assigned</b>	<b>Marks Weightage %</b>
<b>UNIT-I</b>	1	<b>Introduction to Python Programming Language:</b> Understandingpython blocks, Introduction to Python Language,Installation and Working with Pytho, Understanding Python variables, Python basic Operators, Strengths andWeaknesses	8	16
	2	<b>Python Data Types:</b> Declaring and using Numeric data types: int, float, complex, Using string data type and string operations, Defining list and list slicing,Use of Tuple data type, Conversions, Built InFunctions		
<b>UNIT-II</b>	3	<b>PythonProgramFlowControl</b> Conditional blocks using if, else and elif, Simple for loops in python, For loop using ranges, string, list and dictionaries, Use of while loops in python, Loop manipulation using pass, continue, break and else, Programming using Python conditional and loops block	10	20
<b>UNIT-III</b>	4	<b>Functions and Modules :</b> Organizing python codes using functions, Organizing python projects into modules, Importing own module as well as external modules, Understanding PackagesPowerful Lamda function in python, Programming using functions, modules and external packages	16	32

	5	<b>Python File Operation:</b> Reading config files in python, Writing log files in python, Understanding read functions, read(), readline() and readlines(), Understanding write functions, write() and writelines()		
UNIT-IV	6	<b>Creating the GUI Form and Adding Widgets:</b> Widgets: Button, Canvas, Checkbutton, Entry, Frame, Label, Listbox, Menubutton, Menu, Message, Radiobutton, Scale, Scrollbar, text, Toplevel, spinbox, PanedWindow, LabelFrame, tkinterMessagebox. Handling Standard attributes and Properties of Widgets.	16	32
	7	<b>Layout Management:</b> Designing GUI applications with proper Layout Management features.		
<b>Total</b>			<b>50</b>	<b>100</b>

**Text Book:**

1. Learning With Python, by Allen Downey, Jeff Elkner and Chris Meyers

**References:**

1. Dive into Python, Mike
2. Learning Python, 4th Edition by Mark Lutz
3. Programming Python, 4<sup>th</sup> Edition by MarkLutz
4. Python Cookbook, Third edition by David Beazley and Brian K. Jones
5. Head First Python: A Brain-Friendly Guide, by Paul Barry
6. Learn Python The Hard Way, by Zed A. Shaw

<b>Branch: BCA</b>	<b>Semester-IV</b>
<b>Subject Code: 4201</b>	<b>Lecture: 04 Credit: 02</b>
<b>Course Opted</b>	<b>Core Course Practical - 11</b>
<b>Subject Title</b>	<b>PYTHON PROGRAMMING - LAB</b>

**Course objectives:**

- Install and run the Python interpreter
- Gain knowledge of Python syntax
- Learn variable declarations in Python
- Learn control structures
- Understand modules

**Course Outcomes:**

- Will be able to interpret the fundamental Python syntax use Python control flow statements.
- Enable the handling of strings and functions.
- Will be able to create and run Python programs by utilizing the data structures like lists, dictionaries, tuples and sets.

<b>Modules</b>	<b>Sr. No.</b>	<b>Topic and Details</b>	<b>No. of Practicals Assigned</b>	<b>Marks Weightage %</b>
<b>UNIT-I</b>	1	<b>Installation of Python</b> Implementation of Numeric int, float, complex data types and type conversions Implementation of string data type, list, data type Conversions, Use of Built in Functions	4	8
<b>UNIT-II</b>	2	Programs to Implement if, else and elif, loops, Loop manipulation using pass, continue, break and else and loops block	5	10
<b>UNIT-III</b>	3	Programs using functions, modules and external packages Programs to implement file operations	8	16
<b>UNIT-IV</b>	4	Creation of GUI Form and Adding Widgets: Designing GUI applications with proper Layout Management features.	8	16
<b>Total</b>			<b>25</b>	<b>50</b>

**Text Book:**

1. Learning With Python, by Allen Downey, Jeff Elkner and Chris Meyers

**References:**

1. Dive into Python, Mike
2. Learning Python, 4th Edition by Mark Lutz
3. Programming Python, 4<sup>th</sup> Edition by Mark Lutz
4. Python Cookbook, Third edition by David Beazley and Brian K. Jones
5. Head First Python: A Brain-Friendly Guide, by Paul Barry

<b>Branch: BCA</b>	<b>Semester-IV</b>
<b>Subject Code: 4102</b>	<b>Lecture: 04</b> <b>Credit: 04</b>
<b>Course Opted</b>	<b>Core Course - 12</b>
<b>Subject Title</b>	<b>INTRODUCTION TO MICROPROCESSORS</b>

**Course Objectives:**

- To learn and understand technical aspect of 8085 microprocessor.
- To understand the standard instruction set available for 8085 IC.
- To Design and develop various assembly language programs for 8085 IC and 8255 PPL.
- To learn the concept of interrupts
- To understand serial communication and interfacing.
- To understand advance microprocessor 8088/8086.

**Course Outcomes:**

- Understand the architecture and addressing modes of 8085 microprocessor and memory organization and its Interfacing.
- Understand various types of instructions and Instruction Cycled with proper timing diagrams.
- Develop various assembly language programs by using different types of instructions and understand PPL interfacing.
- Understand 8259 interrupt controller IC with its internal organization and single and cascade operation.
- To understand 8086/8088 microprocessor, architecture, instruction set, addressing modes, simple programs, memory organization and interfacing.

<b>Modules</b>	<b>Sr. No.</b>	<b>Topic and Details</b>	<b>No. of Lectures Assigned</b>	<b>Marks Weightage %</b>
<b>UNIT-I</b>	1	<b>8086 Architecture:</b> 8086 Architecture-Functional diagram, Register Organization, Memory Segmentation, Programming Model, Memory addresses, Physical Memory Organization, Architecture of 8086, Signal description of 8086, interrupts of 8086.	8	16
<b>UNIT-II</b>	2	<b>Instruction Set and Assembly Language Programming of 8086:</b> Instruction formats, Addressing modes, Instruction Set, Assembler Directives, Macros, and Simple Programs involving Logical, Branch and Call Instructions, Sorting, String Manipulations	8	16
<b>UNIT-III</b>	3	<b>Programming in 8085:</b> Assembly language programming using 8085, 8255PPL and its interfacing	8	16
	4	<b>Interrupts:</b> Introduction, purpose of interrupts, Interrupt vectors, 8259-Interrupt Controller, Internal organization, pin out, Single and cascaded operation	8	16

<b>UNIT-IV</b>	5	<b>I/O Interface:</b> Serial data transmission, USART 8251 & its organization & interfacing with 8085, 8259 interrupt controller, its organization & interfacing with 8085, DMA controller 8257 & its organization.	8	16
	6	<b>Advance Microprocessor:</b> 8086/8088 microprocessor, architecture, instruction set, addressing modes, simple programs, memory organization and interfacing.	10	20
<b>Total</b>			<b>50</b>	<b>100</b>

**Textbooks:**

1. Microprocessor Architecture Programming ~ Application, with 8080/8085 by Ramesh S. Gaonkar.

**References:**

1. Microprocessor and Digital Systems by D.V.Hall.
2. 16 bit Microprocessor by Triebel and A. Singh.
3. 16 bit microprocessor by Liu and Gibson.

<b>Branch: BCA</b>	<b>Semester-IV</b>
<b>Subject Code: 4202</b>	<b>Lecture: 04</b> <b>Credit: 02</b>
<b>Course Opted</b>	<b>Core Course Practical - 12</b>
<b>Subject Title</b>	<b>INTRODUCTION TO MICROPROCESSORS - LAB</b>

**Course Objectives:**

- To become familiar with the architecture and Instruction set of Intel 8085 microprocessor..
- To be able to develop simple assembly level programs

**Course Outcomes:**

- The student will be familiar with the architecture and Instruction set of Intel 8085 microprocessor
- Will be able to implement assembly level programs

<b>Modules</b>	<b>Sr. No.</b>	<b>Topic and Details</b>	<b>No. of Practicals Assigned</b>	<b>Marks Weightage %</b>
<b>UNIT-I</b>	1	Program to find addition of two 8-bit numbers	1	2
	2	Program to find subtraction of two 8-bit numbers	1	2
	3	Program to find addition of two 16-bit numbers	1	2
	4	Programs to find subtraction of two 16-bit numbers	1	2
<b>UNIT-II</b>	5	Program to find addition of two 16-bit BCD numbers	1	2
	6	Program to find subtraction of two 8-bit BCD numbers	1	2
	7	Program to find multiplication of two 8-bit numbers using successive addition method.	1	2
	8	Program to find multiplication of two 8-bit numbers using shift and add method	1	2
	9	Program to divide 16-bit number by an 8-bit number	1	2
<b>UNIT-III</b>	10	Program to transfer a block of N bytes from source to destination	1	2
	11	Program to find Maximum number in an array	1	2
	12	Program to find Minimum number in an array	1	2
	13	Program to sort the numbers in ascending order	1	2
	14	Program to sort the numbers in descending order	1	2
	15	Program to convert two BCD numbers to their HEX equivalent	1	2
<b>UNIT-IV</b>	16	Program to convert HEX number to BCD	1	2

17	Write a sub routine for 8085 to generate delay of 100 ms	1	2
18	Write a sub routine for 8085 to generate delay of 10 ms	2	4
19	Write a program to generate Fibonacci series	2	4
20	To interface DAC with 8085 to demonstrate the generation of square, saw tooth and triangular wave.	2	4
21	Serial communication between two 8085 through RS-232 C port.	2	4
<b>Total</b>		<b>25</b>	<b>50</b>

**Textbooks:**

1. Microprocessor Architecture Programming ~ Application, with 8080/8085 by Ramesh S. Gaonkar.

**References:**

1. Microprocessor and Digital Systems by D.V.Hall.
2. 16 bit Microprocessor by Triebel and A. Singh.
3. 16 bit microprocessor by Liu and Gibson.

<b>Branch: BCA</b>	<b>Semester-IV</b>
<b>Subject Code: 4103</b>	<b>Lecture: 04 Credit: 04</b>
<b>Course Opted</b>	<b>Core Course - 13</b>
<b>Subject Title</b>	<b>COMPUTER NETWORKS</b>

**Course Objectives:**

- To study TCP/IP & OSI protocol suites
- Learn how computer network hardware and software operate
- Investigate the fundamental issues of network design
- Learn about dominant network technologies

**Course Outcomes:**

- Distinguish between analog and digital signals and understand their characteristics
- Understand the basic concepts of data communications.
- Understand the purpose of network layered models, network communication using the layered concept, and able to compare and contrast Open System Interconnect (OSI) and the Internet Model.
- Understand basic computer network technology.
- Identify the different types of network topologies and protocols.

<b>Module</b>	<b>Sr. No.</b>	<b>Topic and Details</b>	<b>No of Lectures Assigned</b>	<b>Marks Weightage %</b>
<b>UNIT-I</b>	1	<b>Introduction to Networking:</b> Introduction to computer network, network application, network software and hardware components (Interconnection networking devices), Network topology, protocol hierarchies, design issues for the layers, connection oriented and connectionless services.	4	15
	2	<b>Reference models:</b> Layer details of OSI, TCP/IP models. Communication between layer.	4	
<b>UNIT-II</b>	3	<b>Physical Layer:</b> Data and signals, periodic analog signals, digital signals, transmission impairment, data rate limits, performance.	5	5
	4	<b>Digital and Analog transmission:</b> Digital-to-digital conversion, analog-to-digital conversion, transmission modes, digital-to-analog conversion, analog-to-analog conversion.	5	10
	5	<b>Multiplexing, Transmission Media and Switching:</b> Multiplexing – FDM, WDM and TDM Transmission Media – Guided Media (Twisted Pair, Coaxial and Fibre Optics) and Unguided Media i.e. Wireless Media (Radio waves, Microwave, Bluetooth, Infrared) Switching – Circuit and Packet Switching.	5	10



<b>UNIT-III</b>	6	<b>Data link Layer:</b> DLL Design Issues (Services, Framing, Error Control, Flow Control), Error Detection and Correction(Hamming Code, CRC, Checksum), Elementary Data Link protocols, Stop and Wait, Sliding Window(Go Back N, Selective Repeat).	5	10
	7	<b>Medium Access Protocols:</b> Channel Allocation problem, Multiple access Protocol(Aloha, Carrier Sense Multiple Access (CSMA/CD)).	4	10
<b>UNIT-IV</b>	8	<b>Network Layer :</b> Network Layer design issues, Communication Primitives: Unicast, Multicast, Broadcast. IPv4 Addressing (classfull and classless) <b>Routing algorithms :</b> Shortest Path (Dijkstra's), Link State Routing, Distance Vector Routing (Bellmen- Ford) <b>Congestion control algorithms:</b> Open loop congestion control, Closed loop congestion control, Token & Leaky bucket algorithms <b>Virtual Network:</b> VPN, VDA and Cloud Model	8	20
	9	<b>Transport Layer :</b> Introduction, Transport layer protocols (Simple protocol, Stop-and-wait protocol, Go-Back-n protocol, Selective repeat protocol, Bidirectional protocols), Transport layer Services. TCP Segment Format, TCP State Machine, The User Datagram Protocol (UDP).	5	10
	10	<b>Application Layer :</b> Introduction, Services, Architecture, Client Server Model Protocols : DNS, HTTP, SMTP, FTP, Telnet	5	10
<b>Total</b>			<b>50</b>	<b>100</b>

**Text Books:**

1. A.S. Tannenbaum ,”Computer Networks”, 4th edition Prentice hall of India

**References:**

1. Internetworking with TCO/IP: Principles and Architecture, 5th Edition
2. TCP/IP illustrated volume 1 , W. Richard Steven , Addison Westey.
3. Dougals Comer, Computer Networks and Internets , 4 th Edition

<b>Branch: BCA</b>	<b>Semester-IV</b>
<b>Subject Code: 4104</b>	<b>Lecture: 04 Credit: 04</b>
<b>Course Opted</b>	<b>Core Course - 14</b>
<b>Subject Title</b>	<b>SOFTWARE ENGINEERING</b>

**Course Objectives:**

- To understand the nature of software development and software life cycle process models, agile practices.
- To Explain methods of capturing, specifying, visualizing and analyzing software requirements.
- To understand concepts and principles of software design and user-centric approach and principles of effective user interfaces.
- To understand need of project management and project management life cycle.
- To understand project scheduling concept and risk.

**Course Outcomes:**

- Provide the ability to select and apply the knowledge of defined engineering technology activities.
- Able to describe key activities in software development and the role of modeling.
- Able to function effectively on a team whose members together provide leadership, create a collaborative and inclusive environment, establish goals, plan tasks, and meet objectives.
- Able to explain key concepts in software development such as change management, testing and quality.

<b>Module</b>	<b>Sr. No.</b>	<b>Topic and Details</b>	<b>No of Lectures Assigned</b>	<b>Marks Weight age %</b>
<b>UNIT-I</b>	1	<b>Introduction to Software Engineering:</b> Definition, need, software engineering methods, Tools, procedures, Software engineering layers, SEI- CMM, process framework	6	12
	2	<b>Software Process Model:</b> waterfall, spiral, iterative, enhancement and phased development, RAD model, Prototyping model, V life cycle	7	14
	3	<b>Software project planning:</b> Overview, objectives, scope, resources.	2	4
<b>UNIT-II</b>	4	<b>Requirement analysis:</b> Requirement Engineering, Requirements Elicitation and Analysis model, QFD, FAST & Use-cases, Requirements Specification	5	10
	5	<b>Software Project Scheduling:</b> Importance, Work breakdown structure, effort distribution, Gantt chart & tracking the schedule. scheduling Methods: PERT and CPM	5	10
	6	<b>Software Metrics:</b> Direct and indirect methods, size oriented and function oriented metrics, COCOMO	3	6

UNIT-III	7	<b>Design representations:</b> Flow charts, pseudo code, HIPO, DFD, Decision Table <b>Modular design:</b> coupling and cohesion, its various types	5	10
	8	<b>Project Management Spectrum</b> Software Quality assurance: overview, definitions for software quality, roles, verification and validation, FTR, ISO	4	8
	9	<b>Software configuration management:</b> identification, control, auditing, status accounting	2	4
UNIT-IV	10	<b>Software testing Overview:</b> Strategy for testing, White Box Teasting& Black box testing, cyclomatic complexity, Debugging	6	12
	11	<b>Agile Software Development:</b> Coping with Change, The Agile Manifesto:Values and Principles, agile project management	5	10
<b>Total</b>			<b>50</b>	<b>100</b>

**Text Book:**

1. Pressman ,“Software Engineering A Practitioner’s Approach” McGraw-Hill, 5th Edition, 2005

**References:**

1. Shooman, “Software Engineering Design, Reliability and Management” McGraw Hill 198
2. Fairley “Software Engineering Concepts” ” McGraw--Hill Series, New York,