VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY HYDERABAD

B.TECH. MINOR IN INTERNET OF THINGS

COURSE STRUCTURE AND SYLLABUS

(*Applicable for the batches admitted from the academic year 2022-2023*)

V SEMESTER **R22**

Course Code	Title of the Course	L	T	P/D	CH	C
22MC1IN301	Programming in Python	3	0	0	3	3
22MC2IN301	Programming in Python Laboratory	0	0	3	3	1.5
Total		**3**	**0**	**3**	**6**	**4.5**

VI SEMESTER **R22**

Course Code	Title of the Course	L	T	P/D	CH	C
22MC1IN302	IoT Fundamentals and Architecture	3	0	0	3	3
22MC1IN303	Smart Technologies	3	0	0	3	3
Total		**6**	**0**	**0**	**6**	**6**

VII SEMESTER **R22**

Course Code	Title of the Course	L	T	P/D	CH	C
22MC1IN401	Programming Languages for IoT	3	0	0	3	3
22MC2IN401	IoT Automation using Raspberry – PI Laboratory	0	0	3	3	1.5
Total		**3**	**0**	**3**	**6**	**4.5**

VIII SEMESTER **R22**

Course Code	Title of the Course	L	T	P/D	CH	C
22MC1IN402	Fog and Edge Computing for IoT	3	0	0	3	3
Total		**3**	**0**	**0**	**3**	**3**

L – Lecture T – Tutorial P – Practical D – Drawing CH – Contact Hours/Week

C – Credits SE – Sessional Examination CA – Class Assessment ELA – Experiential Learning Assessment

SEE – Semester End Examination D-D – Day to Day Evaluation LR – Lab Record

CP – Course Project PE – Practical Examination

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. Minor (IOT) V Semester

(22MC1IN301) PROGRAMMING IN PYTHON

TEACHING SCHEME		
L	T/P	C
3	0	3

EVALUATION SCHEME				
SE	CA	ELA	SEE	TOTAL
30	5	5	60	100

COURSE OBJECTIVES:

- To learn syntax and semantics and create functions in Python
- To handle strings and files in Python
- To understand lists, dictionaries and regular expressions in Python
- To implement object-oriented programming concepts in Python
- To gain knowledge on web services, network programming and database programming in Python

COURSE OUTCOMES: After completion of the course, the student should be able to

CO-1: Examine Python syntax and semantics and be fluent in the use of Python flow control and functions

CO-2: Demonstrate proficiency in handling strings and file systems

CO-3: Create, run, and manipulate Python programs using core data structures like lists, dictionaries and use regular expressions

CO-4: Interpret the concepts of object-oriented programming as used in Python

CO-5: Implement exemplary applications related to network programming, web services and databases in Python

UNIT – I:

Python: Basics, Objects- Python Objects, Standard Types, Other Built-in Types, Internal Types, Standard Type Operators, Standard Type Built-in Functions, Categorizing the Standard Types, Unsupported Types

Numbers: Introduction to Numbers, Integers, Floating Point Real Numbers, Complex Numbers, Operators, Built-in Functions, Related Modules

Sequences: Strings, Lists, and Tuples, Mapping and Set Types

UNIT – II:

Files: File Objects, File Built-in Function [open()], File Built-in Methods, File Built-in Attributes, Standard Files, Command line Arguments, File System, File Execution, Persistent Storage Modules, Related Modules

Exceptions: Exceptions in Python, Detecting and Handling Exceptions, Context Management, *Exceptions as Strings, Raising Exceptions, Assertions, Standard Exceptions, *Creating Exceptions, Why Exceptions (Now)?, Why Exceptions at All?, Exceptions and the sys Module, Related Modules

Modules: Modules and Files, Namespaces, Importing Modules, Importing Module Attributes, Module Built-in Functions, Packages, Other Features of Modules.

UNIT – III:

Regular Expressions: Introduction, Special Symbols and Characters, Res and Python

Multithreaded Programming: Introduction, Threads and Processes, Python, Threads, and the Global Interpreter Lock, Thread Module, Threading Module, Related Modules

UNIT – IV:

GUI Programming: Introduction, Tkinter programming, Tkinter widgets, Frame, Button, Label, Entry, Brief Tour of Other GUIs.

Web Programming: Introduction, Architecture, CGI environment variable, GET and POST methods, Cookies, File upload, Web Surfing with Python, Creating Simple Web Clients, Advanced Web Clients, CGI-Helping Servers Process Client Data, Building CGI Application, Advanced CGI, Web (HTTP) Servers

UNIT – V:

Database Programming: Introduction, Python Database Application Programmer's Interface (DB-API), Object Relational Managers (ORMs), Related Modules.

TEXT BOOKS:

1. Core Python Programming, Wesley J. Chun, 2nd Edition, Pearson
2. Data Structures and Algorithm Analysis in C, Mark Allen Weiss, 2nd Edition, Pearson Education, 2010
3. Think Python: How to Think Like a Computer Scientist, Allen B. Downey, 2nd Edition, Updated for Python 3, Shroff/O'Reilly Publishers, 2016 (http://greenteapress.com/wp/thinkpython/)

REFERENCES:

1. Data Structures and Algorithms, A. V. Aho, J. E. Hopcroft, and J. D. Ullman, 1st Edition Reprint, Pearson Education, 2003
2. Introduction to Computer Science using Python: A Computational Problem-Solving Focus, Charles Dierbach, Wiley India Edition, 2013
3. Practical Programming: An Introduction to Computer Science using Python 3, Paul Gries, Jennifer Campbell and Jason Montojo, 2nd Edition, Pragmatic Programmers, LLC, 2013

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. Minor (IOT) V Semester

(22MC2IN301) PROGRAMMING IN PYTHON LABORATORY

TEACHING SCHEME		
L	T/P	C
0	3	1.5

EVALUATION SCHEME					
D-D	PE	LR	CP	SEE	TOTAL
10	10	10	10	60	100

COURSE OBJECTIVES:

- To understand program design with functions using Python programming language
- To understand a range of object-oriented programming, as well as in-depth data and information processing techniques
- To understand web scraping using Python
- To understand the high-performance programs designed to strengthen the practical expertise.
- To gain knowledge on Turtle graphics library

COURSE OUTCOMES: After completion of the course, the student should be able to

CO-1: Develop the python scripts using regular expressions

CO-2: Explore python especially the object-oriented concepts, and the built-in objects of Python

CO-3: Develop GUI Application using Python tkinter

CO-4: Create practical and contemporary applications such as web applications, discrete event simulations

CO-5: Implement python program using the Turtle graphics library.

LIST OF PROGRAMS:

1. Python program to calculate student grade.
2. Python example to implement destructor and constructors using __del__() and __init__()
3. Python example to implement Getters and Setters in a class
4. Python - Implement Abstraction using Abstract class
5. Python - Implement Interface using class
6. Python - Create Employee Class with Constructor and Destructor
7. Example of single inheritance in Python
8. Python program to illustrate Single Inheritance
9. Example of multiple inheritance in Python
10. Example of Multilevel Inheritance in Python
11. Example of Hierarchical Inheritance in Python
12. Using Regular Expressions, develop a Python program to

a) Identify a word with a sequence of one upper case letter followed by lower case letters.

b) Find all the patterns of "1(0+)1" in a given string.

c) Match a word containing 'z' followed by one or more o's. Prompt the user for input.

13. GUI&DB

a) Design a GUI based calculator to perform arithmetic operations like addition, subtraction, multiplication and division. (Hint: Expression Calculator using tk)
b) Design a GUI based application to convert temperature from Celsius to Fahrenheit.
c) Write a python program to perform various database operations (create, insert, delete, update).

14. GRAPHICS

a) Consider turtle object. Write functions to draw triangle, rectangle, polygon, circle and sphere. Use object-oriented approach.
b) Design a Python program using the Turtle graphics library to construct a turtle bar chart representing the grades obtained by N students read from a file categorizing them into distinction, first class, second class, third class and failed.

TEXT BOOKS:

1. Core Python Programming, Wesley J. Chun, 2nd Edition, Pearson
2. Data Structures and Algorithm Analysis in C, Mark Allen Weiss, 2nd Edition, Pearson Education, 2010
3. Think Python: How to Think Like a Computer Scientist, Allen B. Downey, 2nd Edition, Updated for Python 3, Shroff/O'Reilly Publishers, 2016 (http://greenteapress.com/wp/thinkpython/)

REFERENCES:

1. Data Structures and Algorithms, A. V. Aho, J. E. Hopcroft, and J. D. Ullman, 1st Edition Reprint, Pearson Education, 2003
2. Introduction to Computer Science using Python: A Computational Problem-Solving Focus, Charles Dierbach, Wiley India Edition, 2013
3. Practical Programming: An Introduction to Computer Science using Python 3, Paul Gries, Jennifer Campbell and Jason Montojo, 2nd Edition, Pragmatic Programmers, LLC, 2013

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. Minor (IOT) VI Semester

(22MC1IN302) IOT FUNDAMENTALS AND ARCHITECTURE

TEACHING SCHEME		
L	T/P	C
3	1	4

EVALUATION SCHEME				
SE	CA	ELA	SEE	TOTAL
30	5	5	60	100

COURSE OBJECTIVES:

- To introduce terminology, technology and its applications
- To learn Raspberry PI platform, that is widely used in IoT applications
- To understand the implementation of web-based services on IoT devices
- To gain knowledge on the internet of things and its characteristics
- To learn concepts on API web servers

COURSE OUTCOMES: After completion of the course, the student should be able to

CO-1: Understand IoT value chain structure (device, data cloud), application areas and technologies involved

CO-2: Understand IoT sensors and technological challenges faced by IoT devices, with a focus on wireless, energy, power, and sensing modules

CO-3: Implement basic IoT applications on embedded platform

CO-4: Understand building blocks of Internet of Things and characteristics

CO-5: Analyze the concepts on API web servers.

UNIT – I:

Introduction to Internet of Things: Definition and Characteristics of IoT, Sensors, Actuators, Physical Design of IoT – IoT Protocols, IoT communication models, IoT Communication APIs, IoT enabled Technologies – Wireless Sensor Networks, Cloud Computing, Embedded Systems, IoT Levels and Templates, Domain Specific IoTs – Home, City, Environment, Energy, Agriculture and Industry.

UNIT – II:

IoT and M2M: Software defined networks, network function virtualization, difference between SDN and NFV for IoT, Basics of IoT System Management with NETCOZF, YANG-NETCONF, YANG, SNMP NETOPEER

UNIT – III:

IoT Physical Devices and Endpoints: Introduction to Arduino and Raspberry Pi-Installation, Interfaces (serial, SPI, I2C), Programming – Python program with Raspberry PI with focus on interfacing external gadgets, controlling output, reading input from pins.

UNIT – IV:

Controlling Hardware: Connecting LED, Buzzer, Switching High Power devices with transistors, Controlling AC Power devices with Relays, Controlling servo motor, speed control of DC Motor, unipolar and bipolar Stepper motors.

Sensors: Light sensor, temperature sensor with thermistor, voltage sensor, ADC and DAC, Temperature and Humidity Sensor DHT11, Motion Detection Sensors, Wireless Bluetooth Sensors, Level Sensors, USB Sensors, Embedded Sensors, Distance Measurement with ultrasound sensor

UNIT – V:

IoT Physical Servers and Cloud Offerings: Introduction to Cloud Storage models and communication APIs Webserver – Web server for IoT, Cloud for IoT, Python web application framework Designing a RESTful web API

TEXT BOOKS:

1. Internet of Things - A Hands-on Approach, Arshdeep Bahga and Vijay Madisetti, Universities Press, 2015
2. Getting Started with Raspberry Pi, Matt Richardson & Shawn Wallace, O'Reilly (SPD), 2014
3. Raspberry Pi Cookbook, Software and Hardware Problems and Solutions, Simon Monk, O'Reilly (SPD), 2016

REFERENCES:

1. Learning Internet of Things, Peter Waher, Ovidiu Vermesan, Packt Publishing, 2015
2. Internet of Things – From Research and Innovation to Market Deployment, Peter Friess, River Publishers, 2014
3. Actuators and Their Interfaces, N. Ida, Sensors, SciTech Publishers, 2014

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. Minor (IOT) VI Semester

(22MC1IN303) SMART TECHNOLOGIES

TEACHING SCHEME		
L	T/P	C
3	1	4

EVALUATION SCHEME				
SE	CA	ELA	SEE	TOTAL
30	5	5	60	100

COURSE OBJECTIVES:

- To provide a detailed description of the integral aspects of smart technologies and their evolution to their current state
- To discuss the potential use of Internet of things (IoT) in reducing counterproductive work behaviours and identifying some of the challenges that organizations might face while implementing IoT in its systems
- To present case studies using easy-to-understand language to explain the breadth and scope of application areas
- To learn the knowledge on cyber care
- To understand the importance of NICE labs

COURSE OUTCOMES: After completion of course, the student should be able to

CO-1: Explain importance, scope and applications of smart technologies

CO-2: Implement drones an IoT based services and blockchain technologies

CO-3: Provide solutions for real world IoT applications like vehicle cyber security

CO-4: Understand the importance of cyber care and develop IoT-Based smart farming solutions

CO-5: Analyze case study on NICE Labs

UNIT – I:

Smart Technologies: Scope and Applications, Cutting-Edge Digitization Challenges in Vehicle Cyber Physical Systems and Cybersecurity, Big Data Analytics as an Enabler in Smart Governance for the Future Smart Cities

UNIT – II:

Digital Masters: Blueprinting Digital Transformation, UAVs/Drones-Based IoT Services, Role of Cyber Security in Drone Technology

Bitcoins as an Implementation of Blockchain and Its Convergence with Internet of Things, Tomorrow's AI-Enabled Banking, Exploring Connected Cars

UNIT – III:

Vehicular Cybersecurity Through Intrusion Detection and Prevention Architecture, Mechanism Protecting Vehicle-to-Vehicle Communication, Advanced Driver Assistance Systems

UNIT – IV:

Cyber Care: Role of Cyber Security in Healthcare Industry, Smart Agriculture: A Tango Between Modern IoT-Based Technologies and Traditional Agriculture Techniques.

UNIT – V:

Importance of Being 'NICE' While Developing IoT-Based Smart Farming Solutions: A Case Study About 'NICE' Labs.

TEXT BOOK:

1. Smart Technologies-Scope and Applications, K. B. Akhilesh, Dietmar P. F. Möller, Springer, 2020