

## L1/L2 NETWORK PROTOCOL TESTING

### MODULE 1 : BASIC OF NETWORKING

- OSI Model
- TCP/IP Layers
- Service data unit & protocol data unit
- Protocols and standards
- Network
- What is network & Internet
- Network core –circuit switching & packet switching
- Network of Networks
- Delay and Throughput in computer network.
- Protocol layered architecture

### MODULE 2 : APPLICATION LAYER

- Network application architecture
- Process communication
- Application layer protocol
- Web & HTTP
- Mail message format
- Mail access protocol
- DNS records and messages
- Video streaming

### MODULE 3 : TRANSPORT LAYER

- Connectionless Transport : UDP
  - a. UDP segment structure and checksum
  - b. UDP checksum
  - c. GBN & SR
- Connection oriented Transport : TCP
  - a. TCP segment structure
  - b. TCP connection management
  - c. TCP congestion control

### MODULE 4 : NETWORK LAYERS

- Router and routing protocol.
- Port processing and forwarding of data.
- Packet scheduling
- Internet protocol : IPV4,IPV6

- SDN
- Routing algorithm
- BGP
- ICMP
- Network management and SNMP

## **MODULE 5 : LINK LAYER AND LAN**

- Multiple access links and process
- Switched local and network
- Link virtualization
- Data center networking

## **MODULE 6 : WIRELESS AND MOBILE NETWORK CDMA**

- WIFI 802.11 wireless LAN
- 802.11 Architecture
- 802.11 MAC protocol
- IEEE 802.11 Frame
- Mobility in same IP subnet
- Personal area network :
  - a. Bluetooth and Zigbee
  - b. Mobility management
  - c. Mobile IP

## **MODULE 7 : NETWORK SECURITY**

- Network security and cryptography
- Message integrity and digital signature
- Authentication protocol
- Securing email
- Securing TCP connection
- Network layer security
- Securing WLAN

## **MODULE 8 : MULTIMEDIA NETWORKING**

- Properties of audio and video
- UDP and HTTP Streaming
- VOIP
- SIP and SDP
- RTP
- Different server

## MODULE 9 : PYTHON PROGRAMMING COURSE CONTENT

### MODULE 9.1: BASICS OF PYTHON

- Introduction

#### *PYTHON OPERATORS*

- Arithmetic
- Relational
- Logical
- Assignment
- Bitwise Membership
- Identity operators.

#### *PYTHON CONDITIONAL STATEMENTS*

- If
- If - else
- If – elif

#### *PYTHON LOOPS*

- While
- For
- Range()
- Break and Continue
- Example problems

#### *PYTHON NUMBERS*

- Types in numbers
- Type conversions

#### *PYTHON STRINGS*

- Built-in functions
- Basic operators
- Slicing
- Example problems

#### *PYTHON LISTS*

- Array or list in python
- List slicing techniques
- Built-in functions

#### *PYTHON TUPLE*

- Tuple & Immutability
- Built-in functions
- Example problems

# Primebit solution

---

## ***PYTHON DICTIONARIES***

- Creation of dictionaries
- Built-in functions
- Example problems

## ***PYTHON FUNCTIONS***

- Advantages of functions
- Function definition using
- def statement.
- Calling function.
- passing parameters
- scope of the variables

## ***MODULES IN PYTHON***

- Importing modules
- creation of own modules
- standard modules - os / sys

## ***EXCEPTION HANDLING***

- Handling exceptions
- Try-except block
- Example problems

## ***FILE HANDLINGS***

- Files and directories
- Text files
- Binary files
- Text file processing
- Binary file processing

## **MODULE 9.2 : ADVANCED PYTHON**

### ***NETWORK PROGRAMMING:***

- In this, we will teach how we will send as well as receive data between devices by using TCP/IP and UDP protocols.
- Python supports many networking protocols through libraries such as a socket, SMTP etc and you can also write network programs directly by using TCP/IP or UDP sockets.

### ***TCP VS UDP / CLIENT VS SERVER - PYTHON EXAMPLES:***

- Python socket module makes it easy for you to write our own clients and server programs.

# Primebit solution

---

## ***MULTIPROCESSING IN PYTHON:***

Multiprocessing refers the ability of a system to support more than one processor at the same time. Applications in multiprocessing system are broken to smaller routines that run independently. The operation system allocates these threads to the processors improving performance of the system.

## ***FILE TRANSFER PROTOCOLS:***

In this, we will teach how can we transfer the files as well as receive the files between the devices.

## ***EMAILING:***

To transfer the emails, we will use SMTP library which is a built-in library in python. SMTP ( Simple Mail Transfer Protocol) is a protocol which enables you to send emails.