

5G Testing with Python Programming Course Curriculum

Module 1: 5G N/W Requirements and Design

- Limitation of 4G and challenges for 5G
- Capacity and higher data rate
- QOE (Quality of Experience)
- Spectrum
- mMIMO(Massive multiple Input –Multiple Output)
- Device to Drive Communication
- Reduced Protocol Overhead
- Hydrogenous cell Architecture
- Network function Visualization

Module 2: 5G Software controlled Architecture

- Software in Radio access Network
- Software in mobile edge network
- Software in transport network
- 5G protocol stack Architecture
- Security in 5G Software Network
- Radio Access protocol in cloud RAN

Module 3: 5G Hardware Network Architecture 5G Cloud RAN

- Network Architecture Evolution from 4G to 5G
- End to End Network slicing
- 5G ---Cloud RAN
 - o Cloud RAN Advantage
 - o Advance Cloud RAN Architecture
 - o Cloud RAN and 5g

- o Functional split of Cloud RAN (real time) RT and NRT split
- o MultiRAT architecture for 4G /5G cloud RAN
- o Distributed RAN
- o Virtualization RAN

- Fixed Network

- o Heterogeneous access domain
- o Flexible metro domain
- o Integrations of access ,metro core Technologies

Module 4: 5G Data flow channels and channel coding

- Logical Channel
- Transport Channel
- Physical Channel
- UL Mapping Of channels
- DL- mapping of Channels
- Channel coding
- Multiplexing
- Interleaving

Module 5: 5G Physical layer Design

- Frame structure
- Slot structure and physical resources
- Reference signal
 - o UE specific reference signal XPDSCH
 - o UE Specific Reference signal XPDCCH
 - o CSI reference Signal
 - o Beam Reference signal
 - o Demultiplexing reference signal
- Synchronization signal

- o Primary Synchronization signal
 - o Secondary synchronization signal
 - o Extended synchronisation signal
- 5G Modulation
 - o BPSK—16QAM
 - o QPSK—64 QAM
- Physical channel
 - o Uplink XPUSCH, Uplink XPUCCH, Uplink XPRACH
- Downlink Physical channel
 - o XPDSCH
 - o XPBCH and Extended PBCH
 - o XPDCCH
- Mapping of channel to resource element

Module 6: Physical Layer Procedure

- Synchronization procedure
- Beam Forming Procedure
- Power Control Procedure
- Physical Random Access Procedure

Module 7: 5G MAC Layer Design

- Comparison of 5G MAC with UMTS and LTE MAC
- 5G – MAC Architecture
- 5G –MAC Service
- 5G – MAC Function
- Mapping of Transport channel and logical channel

Module 8: 5G MAC layer procedures

- 5G Random Access Procedure
- XDL SCH Data Transfer
- XUL SCH Data Transfer
- Beam management
- 5G ---MAC Protocol Data Unit
- 5G---MAC format
- 5G---MAC Parameter

Module 9: 5G RLC Layer Design

- Compare LTE , RLC and 5G RLC
- 5G---RLC Architecture
- Model of 5G-RLC sub layer
- RLC mode
- 5G --RLC service
- 5G –RLC function
- Data available for transmission

Module 10: 5G RLC procedures

- Data transfer Procedure
- ARQ Procedure
- SDU Discard procedure
- Reestablishment Procedure
- 5G—RLC protocol data Unit
- 5G RLC Formats and Parameter

Module 11: 5G PDCP layer design

- Compare 5G PDCP with LTE and UMTS PDCP
- 5G PDCP Architecture
- 5G PDCP Service
- 5G Function
- Data Available for Transmissions

Module 12: 5G PDCP layer Procedure

- 5G PDCP Data Transfer procedure
- 5G PDCP States Report
- 5G PDCP Discard
- Ciphering and Deciphering
- Integrity Protection and Verification
- 5G PDCP Protocol data Unit
- 5G PDCP Format
- 5G PDCP Parameters

Module 13: RRC layer Design

- RRC layer Architecture
- RRC States
- Signalling radio bearers
- 5G –RRC services
- 5G—RRC function

Module 14: RRC layer procedures

- Broadcast system information
- RRC connection management
- RRC connection reconfiguration
- RRC connection Reestablishment
- RRC connection release
- RRC measurement procedure
- 5G- RRC Protocol data unit

Module 15: NAS Layer

- Mobility Management
- Session Management function
- 5G-NAS State
- 5G- NAS MM Protocol
- 5G- NAS SM Protocol

Module 16: 5G-Variable, Timer and Constant

- 5G-MAC Variable, timer and constant
- 5G-RLC Variable, timer and constant
- 5G-PDCP Variable, timer and constant
- 5G-RRC Variable, timer and constant

Module 17: 5G-Security design

- 5G Security Network Architecture
- Domains in 5G security
- 5G Security software architecture
- Stratum
- Security control class(SCC)
- Security Realm(SR)
- 5G-Security methods

Python Programming Content

Module 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

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 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 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 program.

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 a smaller routines that run independently. The operation system allocates this 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 built-in library in python. SMTP (Simple Mail Transfer Protocol) is a protocol which is enable you to send emails

Topics Covered In The advanced Python Module

- . Multitasking
 - Multiprocessing
 - Multithreading
- . Network Programming
 - Networking Principles
 - TCP/IP Basics
 - Sockets, The Socket module, Clients and Servers
 - Tcp Client-Server Programming
 - UDP Client-Server Programming
 - Emailing
 - FTP Protocol