# ANNEXURE - IV

# HONORS
# offered by
# Department of Information Technology

# B.Tech – IT (Honors)

| SNo. | Course Name | L-T-P | CR | Prerequisites | Offered to |
|---|---|---|---|---|---|
| **POOL-1** | | | | | |
| 1 | Practical Cryptography in Python | 3-1-0 | 4 | Python | IT |
| 2 | Practical Machine Learning with Python | 3-1-0 | 4 | Python | IT |
| 3 | Parallel Programming | 3-1-0 | 4 | Data structures Programming | IT |
| 4 | Enterprise Architecture Foundations | 3-1-0 | 4 | Software Engineering | IT |
| **POOL-2** | | | | | |
| 1 | Computer Forensics and Cyber Crime | 3-1-0 | 4 | Security | IT |
| 2 | Business Intelligence | 3-1-0 | 4 | Data Science | IT |
| 3 | Distributed and Cloud Computing: From Parallel Processing to IoT | 3-1-0 | 4 | Parallel Programming | IT |
| 4 | Software Design | 3-1-0 | 4 | Software Engineering | IT |
| **POOL-3** | | | | | |
| 1 | Cyber Laws and Ethics | 3-1-0 | 4 | Cyber Crime | IT |
| 2 | Reinforcement Learning | 3-1-0 | 4 | Machine Learning | IT |
| 3 | Cloud Networking | 3-1-0 | 4 | Cloud Computing Networks | IT |
| 4 | Software Quality Analysis | 3-1-0 | 4 | Scripting Languages | IT |
| **POOL-4** | | | | | |
| 1 | Blockchain Fundamentals | 3-1-0 | 4 | Security | IT |
| 2 | Computer Vision | 3-1-0 | 4 | Graphics Programming | IT |
| 3 | Cloud Security and Privacy | 3-1-0 | 4 | Cloud Computing Security | IT |
| 4 | Agile Software Development | 3-1-0 | 4 | Software Engineering | IT |

## Streams

1. Security & Forensics
2. Data Engineering
3. Distributed and Cloud Computing
4. Software Engineering

# PRACTICAL CRYPTOGRAPHY USING PYTHON

**Course Code – Category:**
L T P E O
3 1 0 0 0

**CREDITS 4**
Sessional Marks: 40
End Exam Marks: 60
End Exam: 3 Hours

**Prerequisites: Python programming**

**Course Objective:**

The course is intended to develop a greater intuition for the proper use of cryptography and the basics of writing cryptographic algorithms in Python, demystifies cryptographic internals, and demonstrates common ways cryptography is used incorrectly.

**Course Outcomes:**

After completion of this course, the students will be able to:
**CO-1**. Provide security of the data over the network.
**CO-2**. Implement various networking protocols.
**CO-3**. Protect any network from the threats in the world.
**CO-4**. Do research in the emerging areas of cryptography and network security

**Mapping of Course Outcomes with POs and PSOs**

| COs/POs-PSOs | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | P010 | PO11 | PO12 | PSO1 | PSO2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | | 1 | 2 | 2 |
| CO2 | 1 | 2 | 2 | 1 | | 1 | 1 | 1 | 1 | | | 1 | 3 | 2 |
| CO3 | 1 | 3 | 1 | 1 | 1 | | | 1 | 1 | 1 | | 1 | 3 | 2 |
| CO4 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | | 1 | 3 | 2 |

**UNIT-I: 12 Periods**

**Cryptography: More Than Secrecy**

Setting Up Your Python Environment, Caesar's Shifty Cipher, A Gentle Introduction to Cryptography, Uses of Cryptography, What Could Go Wrong?, YANAC: You Are Not A Cryptographer, "Jump Off This Cliff"—The Internet.

**Hashing**

Hash Liberally with hashlib, Making a Hash of Education, Digestible Hash, Pass Hashword,Hash Passwords, Cracking Weak Passwords.

**Learning Outcomes:** At the end of this unit, the students will be able to

- Understand where cryptography is used, why, and how it gets misused
- Know what secure hashing is used for and its basic properties

**UNIT-II: 12 Periods**

**Symmetric Encryption: Two Sides, One Key** –Let's Scramble!, What Is Encryption, Really?, AES: A Symmetric Block Cipher, ECB Is Not for Me, Wanted: Spontaneous Independence, Key and IV Management, Exploiting Malleability, Weak Keys, Bad Management, finalize().

**Learning Outcomes:** At the end of this unit, the students will be able to

- Get up to speed on algorithms and modes for block ciphers such as AES, and see how bad configurations break

**UNIT-III: 12 Periods**

**Asymmetric Encryption: Public/Private Keys** - A Tale of Two Keys, Getting Keyed Up, RSA Done Wrong: Part One, Stuffing the Outbox, What Makes Asymmetric Encryption Different?, Pass the Padding, The Proof Is in the Padding, Exploiting RSA Encryption with PKCS #1 v1.5 Padding, Additional Notes About RSA,

**Learning Outcomes:**At the end of this unit, the students will be able to

- Discover how RSA encryption can be broken if insecure padding is used

**UNIT-IV: 12 Periods**

**Message Integrity, Signatures, and Certificates-** An Overly Simplistic Message Authentication Code (MAC), MAC, HMAC, and CBC-MAC, Digital Signatures: Authentication and Integrity, Certificates: Proving Ownership of Public Keys, Certificates and Trust, Revocation and Private Key Protection, Replay Attacks, ummarize-Then-MAC

**Learning Outcomes:** At the end of this unit, the students will be able to

- Use message integrity and/or digital signatures to protect messages

**UNIT-V: 12 Periods**

**Combining Asymmetric and Symmetric Algorithms–** Exchange AES Keys with RSA, Asymmetric and Symmetric: Like Chocolate and Peanut Butter, Measuring RSA's Relative Performance, Diffie-Hellman and Key Agreement, Diffie-Hellman and Forward Secrecy, Challenge-Response Protocols,

**More Symmetric Crypto: Authenticated Encryption and Kerberos**-AES-GCM, AES-GCM Details and Nuances, An Introduction to Kerberos

**Learning Outcomes:** At the end of this unit, the students will be able to

- Utilize modern symmetric ciphers such as AES-GCM and CHACHA

**Text Book:s**

1. Practical Cryptography in Python, Learning Correct Cryptography by Example - Authors Seth James Nielson, Christopher K. Monson

**Reference Books:**

1. Practical Cryptography for Developers, Svetlin Nakov, https://cryptobook.nakov.com/ (This book is freely available.)

# PRACTICAL MACHINE LEARNING WITH PYTHON

**Course Code – Category:**

| L | T | P | E | O |
|---|---|---|---|---|
| 3 | 1 | 0 | 0 | 0 |

**CREDITS 4**
Sessional Marks: 40
End Exam Marks: 60
End Exam: 3 Hours

**Prerequisite(s):**

- Python programming knowledge
- Basic knowledge of mathematics and statistics

**Course Objectives**

- Learn the purpose of Machine Learning and where it applies in the real world
- Explore several algorithms and see how they help us perform several machine learning tasks.

**Course Outcomes**

On completing this course student will be able to

**CO-1:** Illustrate basics of machine learning using Python

**CO-2:** Describe supervised and unsupervised algorithms and their practical uses

**CO-3:** Use a suitable programming language to work with data and apply machine learning tools to it.

**CO-4:** Design simple algorithms, code them with python and test with benchmark datasets.

| | PO1 | PO 2 | PO 3 | PO 4 | PO 5 | PO 6 | PO 7 | PO 8 | PO 9 | PO 1 0 | PO 1 1 | PO 1 2 | PSO 1 | PSO 2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | 1 | | | | | | | 1 | 1 | | 2 | 2 | 2 |
| CO2 | 3 | | 1 | 2 | 2 | 2 | | | 1 | 1 | | 2 | 2 | 2 |
| CO3 | 2 | 2 | 2 | 2 | 3 | 2 | 1 | 1 | 3 | 3 | 2 | 3 | 3 | 2 |
| CO 4 | 2 | 2 | 3 | 3 | 3 | 2 | 2 | 1 | 3 | 3 | 2 | 3 | 3 | 2 |

**UNIT-I: Machine Learning Basics** **10 Periods**

Introduction, Why Machine Learning, , Need for Machine Learning, Understanding Machine Learning, Types of Machine Learning, Main Challenges in Machine Learning, Real world applications of Machine Learning

**Learning outcomes: At the end of the unit, the student will be able to**

- Learn why we need machine learning and its fundamentals
- How best we can leverage Machine Learning to get the maximum from your data

**UNIT-II: Python Machine Learning Ecosystem** **10 Periods**

Introduction**,** Strengths**,** Installing Jupyter notebook, numpy, pandas, Scikit- Learn, Essential Libraries and Tools, Loading sample dataset, loading a CSV File, Loading an Excel File, First Application: Classifying iris species, training and Testing Data, Build the model (knn), making predictions, evaluating model, Pre-processing and Feature Extraction

**Learning outcomes: At the end of the unit, the student will be able to**

- Learn why python is apt language for Machine Learning
- Use python libraries to solve machine learning problems

**UNIT-III: Data Wrangling** **11 Periods**

DataCollection, Data Description, Understanding Data, Get Data, Imputing Missing Values, Handling Duplicates, Handling Categorical Data, Normalizing Values, Prepare Data for Machine Learning Algorithms, Data Visualization,.

**Learning outcomes: At the end of the unit, the student will be able to**

- Learn how raw data is converted to a form where it can be used in machine learning algorithms
- Implement Data wrangling techniques to get the data into a form where it can be utilized in Machine Learning Algorithms for analysis

**UNIT-IV: Supervised Learning** **(12 Periods)**

Introduction, Logistic Regression, K nearest Neighbours, Naive Bayes Classifier, Support Vector Machines, Random Forest

**Learning outcomes: At the end of the unit, the student will be able to**

- Identify the characteristics of data and implement techniques to solve problem
- Analyze the data and predict decisions using Supervised Learning algorithms

**UNIT-V: Unsupervised Learning** **(12 periods)**

Introduction, Supervised VS unsupervised, Types of unsupervised Learning, Challenges in Unsupervised Learning K Means Clustering, DBSCAN, Principal Component Analysis

**Learning outcomes: At the end the unit, student will be able to**

- Distinguish supervised and unsupervised learning
- Analyze the data and predict decisions using Unsupervised Learning algorithms

**Textbooks:**

- Practical Machine Learning with Python by Dipanjan Sarkar Raghav Bali Tushar Sharma
- Introduction to Machine Learning with Python by Andreas C. Müller, Sarah Guido

**Reference Books:**

1. Machine Learning with Python Cookbook by Chris Albon
2. Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow by Aurélien Géron

# PARALLEL PROGRAMMING

**Course Code – Category:**
L T P E O
3 1 0 0 0

**CREDITS 4**
Sessional Marks: 40
End Exam Marks: 60
End Exam: 3 Hours

**Prerequisite(s): Data Structures Programming**

## Course Objectives

1. To familiarize the issues in parallel computing.
2. To describe distributed memory programming using MPI.
3. To understand shared memory paradigm with Pthreads and with OpenMP.
4. To learn the GPU based parallel programming using OpenCL.

## Course Outcomes

After completion of this course, the students will be able to:

**CO-1:** Identify issues in parallel programming.

**CO-2:** Develop distributed memory programs using MPI framework.

**CO-3:** Design and develop shared memory parallel programs using Pthreads and using OpenMP.

**CO-4:** Implement Graphical Processing OpenCL programs.

## Mapping of Course Outcomes with POs and PSOs

| COs/POs-PSOs | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 | PSO1 | PSO2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 1 | 3 | 1 | | 2 | | | | 2 | | 3 | 2 | 2 | 2 |
| **CO2** | 2 | | 3 | 2 | 2 | | | | 2 | 1 | | 2 | 3 | 2 |
| **CO3** | 3 | 2 | 3 | 2 | 2 | | | | 2 | 2 | 3 | 2 | 3 | 2 |
| **CO4** | 3 | 2 | 2 | 2 | 2 | | | | 2 | 1 | 3 | 2 | 3 | 2 |

## UNIT-I: FUNDAMENTALS OF PARALLEL COMPUTING 10 Periods

Need for Parallel Computing – Parallel Computer Models – ILP, TLP and Data Parallelism – Parallel Programming Overview – Processes, Tasks and Threads – Parallel Programming Models – Shared Memory Programming – Message Passing Paradigm – Interaction and Communication – Interconnection Networks

**Learning outcomes**: At the end of this unit, the students will be able to

- Explain the different parallel computer models
- Illustrate parallel programming models.
- Describe the interaction and communication inter connection networks

**UNIT-II: CHALLENGES OF PARALLEL PROGRAMMING 8 Periods**

Identifying Potential Parallelism – Techniques for Parallelizing Programs – Issues – Cache Coherence issues – Memory Consistency Models – Maintaining Memory Consistency – Synchronization Issues – Performance Considerations.

**Learning outcomes**: At the end of this unit, the students will be able to

- Classify the techniques for parallelizing programs
- Identify the cache, coherence, and synchronization issues.
- Describe Memory consistency Models
- Identify the performance considerations

**UNIT-III: 8 Periods**

OpenMP Execution Model – Memory Model and Consistency – OpenMP Directives – Run Time Library Routines – Handling Data and Functional Parallelism – Performance Considerations.

**Learning outcomes**: At the end of this unit, the students will be able to

- Explain OpenMP Execution Model – Memory Model and Consistency
- Describe how to handle data functional Parallelism.
- Identify Run Time Library Routines.

**UNIT-IV: 9 Periods**

The MPI Programming Model – MPI Basics – Circuit Satisfiability – Global Operations – Asynchronous Communication – Collective Communication – Other MPI Features – Performance Issues – Combining OpenMP and MPI.

**Learning outcomes**: At the end of this unit, the students will be able to

- Understand the MPI Programming Model.
- Explain Performance issues with MPI programming.

**UNIT-V: PROGRAMMING HETEROGENEOUS PROCESSORS 10 Periods**

GPU Architecture – Basics of CUDA – CUDA Threads – CUDA Memories – Synchronization Handling – Performance Issues – Application Development. Introduction to OpenCL.

**Learning outcomes**: At the end of this unit, the students will be able to

- Explain GPU architecture.
- Describe the CUDA threads and CUDA memories
- Develop the application using OpenCL framework

**TEXT BOOKS**


1. Peter S. Pacheco, "An Introduction to Parallel Programming", Morgan Kaufmann, 2011.
2. Michael J Quinn, "Parallel programming in C with MPI and OpenMP", Tata McGraw Hill, 2003.


**REFERENCES**


1. Ananth Grama, George Karypis, Vipin Kumar and Anshul Gupta, "Introduction to Parallel Computing", Second Edition, Pearson Education Limited, 2003.
2. Shameem Akhter and Jason Roberts, "Multi-core Programming", Intel Press, 2006.
3. Ian Foster, "Designing and Building Parallel Programs: Concepts and Tools for Parallel Software Engineering", Addison Wesley Longman Publishing Co., USA, 1995.
4. David E. Culler, Jaswinder Pal Singh, "Parallel Computing Architecture: A hardware/ Software approach", Morgan Kaufmann / Elsevier Publishers, 1999.
5. OpenMP Programmer's Manual. 6. MPI Programmer's Manual

# Enterprise Architecture Foundation

**Course Code – Category:**

| L | T | P | E | O |
|---|---|---|---|---|
| 3 | 1 | 0 | 0 | 0 |

**CREDITS 4**
Sessional Marks: 40
End Exam Marks: 60
End Exam: 3 Hours

**Prerequisite(s): None**

## Course Objectives

This course covers foundational aspects of both enterprise and architectural thinking, including the software to technology to solution architecture continuum, role of EA in business and IT alignment, architectural styles and techniques for capturing and documenting architectures. Techniques for analyzing and reasoning about architectures are practiced in assignments in class.

## Course Outcomes

After completion of this course, the students will be able to:

**CO-1:** Know theoretical foundations of modern overview, tools and uses of enterprise architecture

**CO-2:** Know main theoretical prerequisites of managing, deploying enterprise architecture

**CO-3:** Formulate documenting enterprise architecture.

**CO-4:** Use methods of interactive modelling of additional enterprise architecture tools.

**Mapping of Course Outcomes with POs and PSOs**

| COs/POs-PSOs | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | P010 | PO11 | PO12 | PSO1 | PSO2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | 3 | 1 | 1 | 1 | 2 | 1 | 1 | 1 | 1 | | 2 | 3 | 2 |
| **CO2** | 3 | 3 | 1 | 1 | 2 | 2 | 1 | 1 | 1 | 1 | | 2 | 3 | 2 |
| **CO3** | 3 | 3 | 1 | 1 | 2 | 2 | 1 | 1 | 1 | 1 | | 2 | 3 | 2 |
| **CO4** | 3 | 3 | 2 | 1 | 2 | 2 | 1 | 1 | 1 | 1 | | 2 | 3 | 2 |

**UNIT-I:** **10 Periods**

**Introduction to EA:** What is Enterprise Architecture,Context for Enterprise Architecture, Levels of Architecture,Types of Architecture, Scope of Architecture, Characteristics of Good Architecture, Lists Diagrams and Matrices.

**Meet the Enterprise Architecture Tools:**Activity Diagram, Auditing, Balanced Scorecard, Business Process Diagram, Calendar, Class Diagram, Component Diagram, Dashboard Diagram, Decision Tree Diagram, Deployment Diagram,Gap Analysis Matrix, Heat Map,

**Learning outcomes**: At the end of this unit, the students will be able to

- Understand the architecture continuum and the relation between software, applications, technology and solution architectures

**UNIT-II:** **10 Periods**

**Meet the Enterprise Architecture Tools:** Import and Export Spreadsheets, Organizational Chart Diagram, Patterns, Relationship Matrix, Requirement Diagram, Roadmap Diagram, Specification Manager, Strategy Map, Team Reviews, Team Reviews, Traceability Window, Value Chain.

**Uses of Enterprise Architecture:** Mergers and Acquisitions, Corporate Divestiture, Architecture Oversight, Business and Systems Improvement, Communication, Enterprise Transitions, Implementation Guide, Portfolio Management.

**Learning outcomes**: At the end of this unit, the students will be able to

- Understand architectural styles and patterns used in solution development and use these in projects
- Understand the concepts and components of business architecture.

**UNIT-III:** **10 Periods**

**Architecture Program Setup:** Management Structure, Architecture Process, Architecture Repository, Tool Setup, Architecture Principles.

**Managing an Enterprise Architecture:** Governance Process, Architecture Steering Committee, Architecture Review Board, Governance Register, Architectures: Strategic Plans, Mission and Vision, Drivers, Goals and Objectives, Capabilities, Business Processes, Conceptual Information Model, Logical Data Model, Schemas and Messages, Physical Data Model, Application Lists Diagrams and Matrices, Application Communication, Interface Lists, Capabilities and Applications, Business Processes and Applications, Application Platform Services, Infrastructure Facilities, Stakeholder Modeling, Requirements Modeling.

**Learning outcomes**: At the end of this unit, the students will be able to

- Use a diagramming tool to develop architectural viewpoints
- Understand and use quality attributes for analyzing and reasoning about architectures

**UNIT-IV:** **10 Periods**

**Document and Enterprise Architecture Techniques:** Architecture Description, Requirement Specification, Vision, Plan, Assessment, Architecture Governance, Requirements Management, Partitioning, Portfolio Management, Balanced Scorecard, Baselines and Versioning, Business Goals and Objectives Modeling, Capability Modeling, Data Modeling, Principles Management, Risk Analysis and Management, Technical Reference Model, Use Cases and Scenarios.

**Learning outcomes**: At the end of this unit, the students will be able to

- Understand the role of enterprise architecture and the path to building enterprise level architecture models.

**UNIT-V:** **10 Periods**

**Additional Enterprise Architecture Tools:** Auto Names and Counters, Baseline Tool, Boundary, Document Artifact, Element Discussions, Glossary, Image Manager, List View, Mind Mapping Diagram, Model Views, Package Browser, Project Browser, Requirements Checklist, Requirement Properties, Risk Taxonomy, Security, Stereotyping, Tagged Values, Visual Filters, Working Sets

**Learning outcomes**: At the end of this unit, the students will be able to

- Effectively participate in a team effort to build architecture for software intensive systems

**TEXT BOOKS**

1. Enterprise Architect User Guide SeriesEnterprise Architecture Author: Sparx Systems & Stephen Maguire Date: 30/06/2017 Version: 1.0
2. An Introduction to Holistic Enterprise Architecture: Fourth Edition 4th Edition, Kindle Edition

**REFERENCES**


1. Designing Enterprise Architecture Frameworks: Integrating Business Processes with IT Infrastructure by N Zarvić, R Wieringa. Apple Academic Press (19 April 2016), 360 p. URL: https://doi.org/10.1201/b16417
2. Neubauer M., Stary CH., S-BPM in the Production Industry. Stakeholder approach, Springer Open, 2017. URL: https://www.springer.com/gp/book/9783319484655


**Note: This course is related to Employability/Skill development.**