

PYTHON PROGRAMMING AND DATA STRUCTURE**Course Code : 314012**

Programme Name/s : Electronics & Computer Engg.
Programme Code : TE
Semester : Fourth
Course Title : PYTHON PROGRAMMING AND DATA STRUCTURE
Course Code : 314012

I. RATIONALE

Python being a powerful programming language with efficient data structures provides an effective approach to Object-oriented programming. Its simplicity and readability make it an excellent language for cultivating problem-solving skills and algorithmic thinking for beginners. Its elegant, simple, and easy-to-understand syntax with its interpreted nature makes it an ideal language for scripting and application development in the field of Electronics and Computer Engineering.

II. INDUSTRY / EMPLOYER EXPECTED OUTCOME

The aim of this course is to help the student to attain the following industry identified competency through various teaching learning experiences:

Develop python program using data structures for given purpose.

III. COURSE LEVEL LEARNING OUTCOMES (COS)

Students will be able to achieve & demonstrate the following COs on completion of course based learning

- CO1 - Develop python program to implement basic building blocks of python.
- CO2 - Perform operations on python data structures
- CO3 - Create modules and packages for given purpose.
- CO4 - Evaluate algorithmic complexity of different searching and sorting algorithms.
- CO5 - Implement Linear Data Structures like stack, queue, linked list using python.

IV. TEACHING-LEARNING & ASSESSMENT SCHEME

| Course Code | Course Title | Abbr | Course Category/s | Learning Scheme | | | | | | Credits | Assessment Scheme | | | | | | | | | | | | | | Total Marks |
|-------------|---------------------------------------|------|-------------------|--------------------------|----|----|----|----|-----|---------|-------------------|-----------|-------|-------|-----|------------------|-----|-------|-----|-------------|-----|-----|-----|-----|-------------|
| | | | | Actual Contact Hrs./Week | | | SL | LH | NLH | | Paper Duration | Theory | | | | Based on LL & TL | | | | Based on SL | | | | | |
| | | | | CL | TL | LL | | | | | | Practical | | | | FA-PR | | SA-PR | | SLA | | | | | |
| | | | | | | | | | | | | FA-TH | SA-TH | Total | | Max | Min | Max | Min | Max | Min | | | | |
| | | | | | | | | | | | | | | Max | Min | | | | | | | Max | Min | Max | |
| 314012 | PYTHON PROGRAMMING AND DATA STRUCTURE | PPP | SEC | 2 | - | 2 | - | 4 | 2 | - | - | - | - | - | 25 | 10 | 50@ | 20 | - | - | 75 | | | | |

PYTHON PROGRAMMING AND DATA STRUCTURE**Course Code : 314012****Total IKS Hrs for Sem. : 0 Hrs**

Abbreviations: CL- Classroom Learning , TL- Tutorial Learning, LL-Laboratory Learning, SLH-Self Learning Hours, NLH-Notional Learning Hours, FA - Formative Assessment, SA -Summative assessment, IKS - Indian Knowledge System, SLA - Self Learning Assessment

Legends: @ Internal Assessment, # External Assessment, *# On Line Examination , @\$ Internal Online Examination

Note :

1. FA-TH represents average of two class tests of 30 marks each conducted during the semester.
2. If candidate is not securing minimum passing marks in FA-PR of any course then the candidate shall be declared as "Detained" in that semester.
3. If candidate is not securing minimum passing marks in SLA of any course then the candidate shall be declared as fail and will have to repeat and resubmit SLA work.
4. Notional Learning hours for the semester are (CL+LL+TL+SL)hrs.* 15 Weeks
5. 1 credit is equivalent to 30 Notional hrs.
6. * Self learning hours shall not be reflected in the Time Table.
7. * Self learning includes micro project / assignment / other activities.

V. THEORY LEARNING OUTCOMES AND ALIGNED COURSE CONTENT

| Sr.No | Theory Learning Outcomes (TLO's) aligned to CO's. | Learning content mapped with Theory Learning Outcomes (TLO's) and CO's. | Suggested Learning Pedagogies. |
|-------|---|--|--|
| 1 | TLO 1.1 Describe basic constructs of Python. TLO 1.2 Write python programs using input output statements with indentation and comments. TLO 1.3 Write python program to evaluate arithmetic expressions. TLO 1.4 Develop programs using Conditional Statements. TLO 1.5 Develop programs using Loop statements. | Unit - I Basic Building blocks in Python 1.1 Introduction to Python- Features of python, Python Identifiers, Keywords, Variables, Constants, Indentation, Comments in python 1.2 Python's Data Types – Numbers, Strings, List, Tuples, Dictionaries, Sets 1.3 Input and output statements 1.4 Operators in Python- Operators as Arithmetic, Assignment, Unary Minus, Relational, Logical, Boolean, Bitwise, Membership, Identity, Operator precedence and Associativity 1.5 Decision Making Statements: - if, if...else, else-if ladder, nested if and switch statement 1.6 Looping statements: - while loop, for loop, nested loop, Manipulating Loops using break, continue and pass statements | Lecture Using Chalk-Board Demonstration Hands-on |

PYTHON PROGRAMMING AND DATA STRUCTURE**Course Code : 314012**

| Sr.No | Theory Learning Outcomes (TLO's) aligned to CO's. | Learning content mapped with Theory Learning Outcomes (TLO's) and CO's. | Suggested Learning Pedagogies. |
|--------------|---|---|---|
| 2 | <p>TLO 2.1 Develop a program to manipulate List for given purpose.</p> <p>TLO 2.2 Develop a program to manipulate Tuples for given purpose.</p> <p>TLO 2.3 Develop a program to manipulate Sets for given purpose.</p> <p>TLO 2.4 Develop a program to manipulate Dictionaries for given purpose.</p> | <p>Unit - II Data Structures in Python</p> <p>2.1 List- Defining List, Creating list, Accessing elements of list, Updating the elements of a list, In built functions for list, List operators- Concatenation of two lists, Repeating of Lists, Membership in list, isOperator, del operator, Aliasing and cloning Lists, List Methods, Nested Lists</p> <p>2.2 Tuples- Defining Tuple, Creating Tuples, Accessing the Tuple elements, Inserting elements in a Tuple, modifying elements of a Tuple, Deleting elements from a Tuple, Basic operations in Tuples, Functions to process Tuples, Nested Tuples</p> <p>2.3 Sets- Defining Set, Creating a Set, Accessing elements from set, Add and update Set, Remove an elements from a Set, Built in functions with Set, Set methods to perform mathematical operations, other relevant set methods</p> <p>2.4 Dictionaries- Defining Dictionary, Creating Dictionary, Accessing elements from Dictionary, Traversing Dictionaries, Add and update Dictionary, Delete an element from a Dictionary, Built in functions of Dictionary, Methods of Dictionary class</p> | <p>Presentations Demonstration Hands-on Flipped Classroom</p> |
| 3 | <p>TLO 3.1 Use python built-in functions.</p> <p>TLO 3.2 Develop relevant user defined function for the given purpose.</p> <p>TLO 3.3 Develop a python module in python for given purpose.</p> <p>TLO 3.4 Develop a python package for given purpose.</p> <p>TLO 3.5 Use NumPy for performing mathematical operations on arrays.</p> <p>TLO 3.6 Develop python program to create objects.</p> | <p>Unit - III Functions, Modules and Packages</p> <p>3.1 Python Functions- Use of python built in functions (e.g. type/data conversion functions, math and string functions), User defined function- Function definition, function calling, function arguments and parameter passing, Return statement, scope of variables (Global and Local Variables)</p> <p>3.2 Modules- Writing modules, importing module, python built in modules (Numeric and mathematical module, Functional Programming Module)</p> <p>3.3 Python packages- Introduction, Writing python packages, using standard packages (NumPy) and user defined package statements</p> <p>3.4 Concept of Classes and Objects- Creating classes and objects in python, Constructors and Destructors in python, Data abstraction and Encapsulation</p> | <p>Presentations Lecture Using Chalk-Board Demonstration Hands-on</p> |

PYTHON PROGRAMMING AND DATA STRUCTURE**Course Code : 314012**

| Sr.No | Theory Learning Outcomes (TLO's) aligned to CO's. | Learning content mapped with Theory Learning Outcomes (TLO's) and CO's. | Suggested Learning Pedagogies. |
|--------------|--|--|--|
| 4 | <p>TLO 4.1 Describe Abstract Data Type (ADT).</p> <p>TLO 4.2 Write an algorithm to search the specific key using given search method.</p> <p>TLO 4.3 Write an algorithm to sort elements using given sorting technique.</p> <p>TLO 4.4 Calculate complexity of given algorithm in terms of Time and space.</p> <p>TLO 4.5 Develop python code to perform searching operation on given array elements.</p> <p>TLO 4.6 Develop python code to perform sorting operation on given array elements.</p> | <p>Unit - IV Fundamentals of Data Structure and Algorithm</p> <p>4.1 Introduction to Data Structure: Concept, Need of Data Structure, Abstract Data Type (ADT), Classification of Data Structures, Operations to be performed on Data Structures, Algorithm, Analysis of algorithm using Time and space Complexity</p> <p>4.2 Searching and Sorting: Linear Search, Binary Search, Bubble Sort, Selection Sort, Insertion Sort</p> | <p>Presentations</p> <p>Lecture Using Chalk-Board</p> <p>Demonstration</p> <p>Hands-on</p> |
| 5 | <p>TLO 5.1 Develop an algorithm to perform PUSH and POP operations for the given stack.</p> <p>TLO 5.2 Develop an algorithm to perform operations on Queue.</p> <p>TLO 5.3 Develop an algorithm to perform operations on singly linked list.</p> | <p>Unit - V Linear Data Structures using Python</p> <p>5.1 Stack: Stack As Abstract Data Type (ADT), Stack Representation in memory using array, Stack Terminologies, Stack Operations- PUSH, POP, Stack Application</p> <p>5.2 Queue: Queue As Abstract Data Type (ADT), Linear Queue Representation in memory using array, Linear Queue Operations- Insertion, Deletion, Queue Application</p> <p>5.3 Linked list: Introduction to Linked List, Singly Linked List representation, Operations on Singly Linked List- Create a linked list, Traverse a linked list, Insertion of a new node in Linked list, Deletion of a node from linked list, Applications of Linked List</p> | <p>Lecture Using Chalk-Board</p> <p>Presentations</p> <p>Demonstration</p> <p>Hands-on</p> |

VI. LABORATORY LEARNING OUTCOME AND ALIGNED PRACTICAL / TUTORIAL EXPERIENCES.

| Practical / Tutorial / Laboratory Learning Outcome (LLO) | Sr No | Laboratory Experiment / Practical Titles / Tutorial Titles | Number of hrs. | Relevant COs |
|--|--------------|--|-----------------------|---------------------|
| <p>LLO 1.1 Install python Integrated Development Environment.</p> <p>LLO 1.2 Write simple python program to evaluate given expression.</p> | 1 | <p>a) Installation of python IDE.</p> <p>b) Develop python program to calculate equivalent registers connected in series and parallel. Accept values of R1, R2 and R3 from the user.</p> <p>c) Develop python program to calculate value of voltage by applying Ohm's law. Accept value of Current(I) and Resistance(R) from the user.</p> | 2 | CO1 |

PYTHON PROGRAMMING AND DATA STRUCTURE**Course Code : 314012**

| Practical / Tutorial / Laboratory Learning Outcome (LLO) | Sr No | Laboratory Experiment / Practical Titles / Tutorial Titles | Number of hrs. | Relevant COs |
|---|--------------|--|-----------------------|---------------------|
| LLO 2.1 Implement two-way branching statement. LLO 2.2 Implement multi-way branching statement. | 2 | a) Develop python program to check whether the entered frequency is radio frequency or audio frequency. b) Develop python program to display various radio frequency bands using if..elseif ladder. c) Develop python program to display resistor color code using switch statement. | 2 | CO1 |
| LLO 3.1 Implement control loops for solving iterative problems. | 3 | *a) Develop python program to demonstrate use of control loops: i) while ii) do while *b) Develop program to demonstrate use of for loop in python (e.g.: various pattern building, printing multiplication table, checking palindrome number etc.) | 2 | CO1 |
| LLO 4.1 Perform basic operations on the Lists in python. | 4 | *Develop python program to perform following operations on List: a) Create b) Access c) Update d) Delete elements from list | 2 | CO2 |
| LLO 5.1 Execute various tuple operations in python. | 5 | *Develop python program to perform following operations on Tuples: a) Create b) Access c) Update d) Delete Tuple elements | 2 | CO2 |
| LLO 6.1 Implement various set operations in python. | 6 | Develop python program to perform following operations on Set: a) Create b) Access c) Update d) Delete Set elements | 2 | CO2 |
| LLO 7.1 Execute various operations on Dictionaries in python. | 7 | Develop python program to perform following operations on Dictionaries in Python: a) Create b) Access c) Update d) Delete e) Looping through Dictionary | 2 | CO2 |
| LLO 8.1 Use built-in mathematical functions and string functions in python. LLO 8.2 Create user defined functions in python. | 8 | a) Develop python program to demonstrate use of math and string built in function. b) Develop python program to implement half adder and full adder by creating function with the help of user defined function. | 2 | CO3 |
| LLO 9.1 Implement class to create object in python. | 9 | Develop python program to define class Student with data members (Roll no, Name, Course, Percentage) as input and then print the details. | 2 | CO3 |

PYTHON PROGRAMMING AND DATA STRUCTURE**Course Code : 314012**

| Practical / Tutorial / Laboratory Learning Outcome (LLO) | Sr No | Laboratory Experiment / Practical Titles / Tutorial Titles | Number of hrs. | Relevant COs |
|--|--------------|--|-----------------------|---------------------|
| LLO 10.1 Use built-in python mathematical modules. LLO 10.2 Write user-defined module in python. | 10 | *a) Develop python program to demonstrate use of: Built-in module (e.g. numeric, mathematical functional and programming module) in python. *b) Develop python program to create a user-defined module (e.g.: building calculator) in python. | 2 | CO3 |
| LLO 11.1 Use python built-in packages. LLO 11.2 Implement user-defined packages in python. | 11 | *a) Develop python program to demonstrate use of NumPy package for creating, accessing and performing different array operations. *b) Develop program to demonstrate the use of user defined packages in python. | 2 | CO3 |
| LLO 12.1 Implement searching techniques in python. | 12 | Implement searching techniques using python a) Linear Search b) Binary Search | 2 | CO4 |
| LLO 13.1 Implement sorting techniques in python. | 13 | *Implement sorting techniques using python a) Bubble Sort b) Selection Sort c) Insertion Sort | 2 | CO4 |
| LLO 14.1 Implement Stack as Abstract Data Type in python. LLO 14.2 Implement Queue as Abstract Data Type in python. | 14 | *a) Develop python program to implement Stack operations (PUSH, POP) using Array. *b) Develop python program to implement Queue operations (enqueue, dequeue) using Array. | 2 | CO5 |
| LLO 15.1 Implement Linked List in python. | 15 | *Develop python program to implement operations (Create, Insert, Delete, Traverse) on Singly Linked List. | 2 | CO5 |
| Note : Out of above suggestive LLOs - <ul style="list-style-type: none"> *' Marked Practicals (LLOs) Are mandatory. Minimum 80% of above list of lab experiment are to be performed. Judicial mix of LLOs are to be performed to achieve desired outcomes. | | | | |

VII. SUGGESTED MICRO PROJECT / ASSIGNMENT/ ACTIVITIES FOR SPECIFIC LEARNING / SKILLS DEVELOPMENT (SELF LEARNING)**Micro project**

- Not Applicable.

Note :

- Above is just a suggestive list of microprojects and assignments; faculty must prepare their own bank of microprojects, assignments, and activities in a similar way.
- The faculty must allocate judicial mix of tasks, considering the weaknesses and / strengths of the student in acquiring the desired skills.
- If a microproject is assigned, it is expected to be completed as a group activity.
- SLA marks shall be awarded as per the continuous assessment record.
- For courses with no SLA component the list of suggestive microprojects / assignments/ activities are optional, faculty may encourage students to perform these tasks for enhanced learning experiences.
- If the course does not have associated SLA component, above suggestive listings is applicable to Tutorials and maybe considered for FA-PR evaluations.

PYTHON PROGRAMMING AND DATA STRUCTURE**Course Code : 314012****VIII. LABORATORY EQUIPMENT / INSTRUMENTS / TOOLS / SOFTWARE REQUIRED**

| Sr.No | Equipment Name with Broad Specifications | Relevant LLO Number |
|-------|---|---------------------|
| 1 | a) Computer System with all necessary peripherals and internet connectivity. b) Any relevant python IDE like IDLE/PyCharm/VSCode/Jupyter Notebook/OnlinePython Compiler. | All |

IX. SUGGESTED WEIGHTAGE TO LEARNING EFFORTS & ASSESSMENT PURPOSE (Specification Table)

| Sr.No | Unit | Unit Title | Aligned COs | Learning Hours | R-Level | U-Level | A-Level | Total Marks |
|--------------------|------|--|-------------|----------------|----------|----------|----------|-------------|
| 1 | I | Basic Building blocks in Python | CO1 | 4 | 0 | 0 | 0 | 0 |
| 2 | II | Data Structures in Python | CO2 | 6 | 0 | 0 | 0 | 0 |
| 3 | III | Functions, Modules and Packages | CO3 | 6 | 0 | 0 | 0 | 0 |
| 4 | IV | Fundamentals of Data Structure and Algorithm | CO4 | 6 | 0 | 0 | 0 | 0 |
| 5 | V | Linear Data Structures using Python | CO5 | 8 | 0 | 0 | 0 | 0 |
| Grand Total | | | | 30 | 0 | 0 | 0 | 0 |

X. ASSESSMENT METHODOLOGIES/TOOLS**Formative assessment (Assessment for Learning)**

- Each practical will be assessed considering – 60% weightage to process and – 40% weightage to product.

Summative Assessment (Assessment of Learning)

- End semester summative assessment of 50 marks for laboratory learning

XI. SUGGESTED COS - POS MATRIX FORM

| Course Outcomes (COs) | Programme Outcomes (POs) | | | | | | | Programme Specific Outcomes* (PSOs) | | |
|-----------------------|--|-----------------------|---------------------------------------|------------------------|--|-------------------------|-------------------------|-------------------------------------|-------|-------|
| | PO-1 Basic and Discipline Specific Knowledge | PO-2 Problem Analysis | PO-3 Design/ Development of Solutions | PO-4 Engineering Tools | PO-5 Engineering Practices for Society, Sustainability and Environment | PO-6 Project Management | PO-7 Life Long Learning | PSO-1 | PSO-2 | PSO-3 |
| CO1 | 1 | 1 | 1 | 1 | | | | | | |
| CO2 | 1 | 1 | 1 | 1 | | | | | | |
| CO3 | 1 | 2 | 2 | 3 | | | 1 | | | |
| CO4 | 1 | 3 | 2 | 1 | | | 1 | | | |
| CO5 | 1 | 2 | 2 | 1 | | | 1 | | | |

Legends :- High:03, Medium:02,Low:01, No Mapping: -
*PSOs are to be formulated at institute level

XII. SUGGESTED LEARNING MATERIALS / BOOKS

| Sr.No | Author | Title | Publisher with ISBN Number |
|-------|--------|-------|----------------------------|
|-------|--------|-------|----------------------------|

PYTHON PROGRAMMING AND DATA STRUCTURE**Course Code : 314012**

| Sr.No | Author | Title | Publisher with ISBN Number |
|-------|--|---|---|
| 1 | David M. Beazley | Python Essential Reference | Addison Wesley, 4th Edition ISBN-13: 978-0-672-32978-4 |
| 2 | Martin C. Brown | The complete Reference Python | Tata McGraw Hill ISBN: 9789387572942 |
| 3 | Dr. Charles R. Severance | Python for Everybody | Shroff Publishers ISBN-13: 978-9352136278 |
| 4 | Mark Lutz | Learning Python | O'Reilly, 4th Edition ISBN: 978-0-596-15806-4 |
| 5 | Ashok Namdev Kamthane Amit Ashok Kamthane | Programming and problem solving with Python | McGraw Hill Education (India) Private Limited ISBN-13: 978-93-87067-58-5 |
| 6 | Michael T. Goodrich Roberto Tamassia Michael H. Goldwasser | Data Structures and Algorithms in Python | WILEY Publication ISBN: 978-1118290279 |
| 7 | Benjamin Baka | Data Structures and Algorithms Using Python | Packt Publishing Ltd. ISBN: 978-1-78646-735-5 |
| 8 | Rance D. Necaise | Data Structures and Algorithms Using Python | WILEY Publication ISBN: 978-0-470-61829-5 |

XIII . LEARNING WEBSITES & PORTALS

| Sr.No | Link / Portal | Description |
|-------|--|--------------------------------------|
| 1 | https://python-iitk.vlabs.ac.in/Introduction.html | Basic constructs of python |
| 2 | https://www.learnpython.org | Python specific data structure |
| 3 | https://www.tutorialspoint.com/python/python_classes_objects.htm | OOP concepts using Python |
| 4 | https://www.pythontutorial.net/python-oop/ | Object Oriented Programming Concepts |
| 5 | https://www.programiz.com/dsa/algorithm https://www.programiz.com/dsa/data-structure-types | Introduction to Data Structure |
| 6 | https://www.programiz.com/dsa/stack https://www.programiz.com/dsa/queue https://www.programiz.com/dsa/linked-list https://www.programiz.com/dsa/linked-list-operations | Linear Data Structure using Python |

Note :

- Teachers are requested to check the creative common license status/financial implications of the suggested online educational resources before use by the students