BACHELOR WITH COMPUTER APPLICATIONS AS MAJOR (CT-1)
6th SEMESTER

CAP622J1: COMPUTER APPLICATIONS _ PYTHON PROGRAMMING

CREDITS: THEORY (3) PRACTICAL (1)

LEARNING OUTCOMES:

After completing this course, the learner shall be able to:

1. *Understand the fundamentals of programming through Python*
2. *Transform a solution from a subjective world into an objective world.*
3. *Work with and manipulate different basic data structures available in Python*
4. *Use the List comprehensions and generators in their programs*
5. *Apply basic object-oriented concepts to design classes and objects in Python*
6. *Use concepts of Inheritance and Polymorphism in their programs*
7. *Perform basic file operations for text and CSV files*
8. *Use existing inbuilt Python modules*
9. *Develop custom modules*
10. *Use basic functionalities provided by packages Numpy and Pandas*
11. *Develop visualizations using different plotting functions available in matplotlib*

UNIT-I: (15 LECTURES)

Origins and History of Python, Structure of a Python Program, Interpreter shell, Indentation, Comments, Identifiers and keywords, Literals, Basic operators (Arithmetic operator, Relational operator, Logical or Boolean operator, Assignment Operator, Bit wise operator)

Entering Expressions into the Interactive Shell, The Integer, Floating-Point, and String Data Types, String Concatenation and Replication, Storing Values in Variables. Creating Python Programs: Input and Output Statements, Control statements: -branching, looping, range function, Exit function, break, continue and pass.

UNIT: II (15 LECTURES)

Functions: Defining a Function, Calling a Function, def Statements with Parameters, Formal and Actual Arguments, Positional Arguments, Keyword Arguments, Default Arguments, Variable Length Arguments, Return Values and return Statements, The None Value, Local and Global Scope, The global Statement.

The List Data Type: Creating Lists, Basic List Operations, Indexing and Slicing in Lists, Built-In Functions Used on Lists, List Methods. List comprehension.

UNIT: III (15 LECTURES)

The Dictionary Data Type: Creating Dictionary, Accessing and Modifying key: value Pairs in Dictionaries, Built-In Functions used on Dictionaries, Dictionary Methods

Tuples and Sets, Creating Tuples, Basic Tuple Operations, Indexing and Slicing in Tuples, Built-In Functions Used on Tuples, Set Methods, Traversing of Sets

Strings, Creating and Storing Strings, Basic String Operations, Accessing Characters in String by Index Number, String Slicing and Joining, String Methods, Formatting Strings.

Files: Types of Files, Creating and Reading Text Data

TEXT BOOKS

1. Gowrishankar S, Veena A, “Introduction to Python Programming”, 1st Edition, CRC Press/Taylor & Francis, 2018. ISBN-13: 978-0815394372
2. Downey, A.B., (2015), Think Python How to think like a Computer Scientist, 3rd edition.
3. Taneja, S. & Kumar, N., (2017), Python Programming- A Modular Approach. Pearson Education.

REFERENCE BOOKS

1. Jake VanderPlas, “Python Data Science Handbook: Essential Tools for Working with Data”, 1st Edition, O'Reilly Media, 2016. ISBN-13: 978-1491912058
2. Aurelien Geron, “Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems”, 2nd Edition, O’Reilly Media, 2019. ISBN – 13: 978-9352139057.
3. Wesley J Chun, “Core Python Applications Programming”, 3rd Edition, Pearson Education India, 2015. ISBN-13: 978-9332555365

4. Miguel Grinberg, “Flask Web Development: Developing Web Applications with Python”, 2nd Edition, O'Reilly Media, 2018. ISBN-13: 978-1491991732.
5. Brown, M. C. (2001). The Complete Reference: Python, McGraw Hill Education.
6. Dromey, R. G. (2006), How to Solve it by Computer, Pearson Education.
7. Guttag, J. V. (2016), Introduction to computation and programming using Python. MIT Press.
8. Liang, Y.D. (2013), Introduction to programming using Python. Pearson Education.

LAB WORK: PYTHON PROGRAMMING

1. Install and configure Python IDE
2. Write simple Python program to display message on screen
3. Write a program to demonstrate different number data types & arithmetic operations in Python.
4. Write a Python program to demonstrate use of conditional statements:
5. Write Python program to demonstrate use of looping statements
6. Demonstrate the following control transfer statements in Python with suitable examples.
    i) break
    ii) continue
    iii) pass
7. Write a python program to demonstrate the working of following functions in Python.
    i) id( )
    ii) type( )
    iii) range( )
8. Write Python programs to demonstrate the following:
    i) input( )
    ii) print( )
    iii) ‘sep’ attribute
    iv) ‘end’ attribute
9. Write a program to create a menu driven program with the following options using functions:
    Addition 2. Subtraction 3. Multiplication 4. Division
10. Write Python program to perform following operations on Lists:
    a) Create list
    b) Access list
    c) Update list (Add item, Remove item)
    d) Delete list
11. Write Python program to perform following operations on Tuples:
    a) Create Tuple
    b) Access Tuple
    c) Update Tuple
    d) Delete Tuple
12. Write Python program to perform following operations on Sets:
    a) Create Set
    b) Access Set elements
    c) Update Set
    d) Delete Set
13. Write Python program to perform following operations on Dictionaries:
    a) Create Dictionary
    b) Access Dictionary elements
    c) Update Dictionary
    d) Delete Set
    e) Looping through Dictionary
14. Write a Python program to demonstrate slicing operations on lists and strings.
15. Write a Python program to perform read and write operations on a file.

BACHELOR WITH COMPUTER APPLICATIONS AS MAJOR (CT-2)

6th SEMESTER

CAP622J2: COMPUTER APPLICATIONS _ COMPUTER ORGANIZATION & ARCHITECTURE

CREDITS: THEORY (4) PRACTICAL (2)

COURSE LEARNING OUTCOMES:

After the course the students are expected to be able to:

- *Explain design of the various functional units and components of computers.*
- *Explain the basics of organizational and architectural issues of a digital computer and Classify and compute the performance of machines.*
- *Explain principles of computer organization and the basic architectural concepts.*
- *Demonstrate memory management and I/O techniques.*
- *To have better idea on how to write assemble language programs.*
- *Explain key aspects of Computer Organization & Architecture by enabling them to perform the experiments with support of a design and simulation.*

UNIT 1: BASIC COMPUTER ORGANISATION (15 HOURS)

Digital Computers, Von Neumann Computer, Basic organisation of a Computer, Computer Generations, Instruction Codes, Stored Program Organization, Computer registers, Computer instructions, Instruction cycle, Fetch and Decode, Common Bus System, Bus Interconnection Structures

UNIT 2: ARITHMETIC LOGIC UNIT (15 HOURS)

General Register Organisation, Control Word, Stack Organisation, Instruction Formats, Addressing Modes (Direct, Indirect, Register, Indexed), Reduced Instruction Set Computer (RISC), Complex Instruction Set Computer (CISC), Instruction Pipelining, Control Unit Implementation: Hardwired and Microprogrammed, Control Memory.

UNIT 3: MEMORY ORGANIZATION (15 HOURS)

Memory Hierarchy, Memory Technology, Main memory, RAM and ROM Chips, Memory Address Map, Memory Connection to CPU. RAM Technologies, SRAM, DRAM, Read Only Memory(ROM), PROM, EEPROM, Flash Memory, Auxiliary memory, Magnetic Disks, Magnetic Tape, Cache memory, Cache Memory Mappings (Direct, Associative, Set-Associative)

UNIT 4: INPUT-OUTPUT ORGANIZATION (15 HOURS)

Peripheral Devices, Input-Output Interface, I/O Bus and Interface Modules, Isolated and Memory Mapped I/O, Modes of Transfer, Programmed I/O, Interrupt-Initiated I/O, Direct Memory Access (DMA), Bus Arbitration, DMA Controller, DMA Transfer, Input-Output Processor (IOP). Introduction to parallel Processing, Multi-processing systems. Multicore architecture.

TEXT BOOK:

1. M.Morris Mano-Computer System Architecture, Revised Third Edition, Pearson Education

REFERENCE BOOK

1. William Stallings-Computer Organization and Architecture, Tenth Edition, Pearson Education
2. Carl Hamacher-Computer Organization, Fifth Edition, Tata McGraw Hill
3. Ramesh S Gaonkar-Microprocessor Architecture, Programming, and Applications with the 8085, Prentice Hall

LIST OF PRACTICALS:

Computer Organization and architecture lab consist of performing various experiments in GNU Sim (An open source and platform independent simulator for 8085 microprocessor).

1. Write the working of 8085 simulator GNUsim8085.
2. Write the working of 8086 simulator EMU8086.
3. Discuss basic architecture of 8085
4. Discuss basic architecture of 8086
5. Discuss the various registers available in 8085.
6. Study the complete instruction set of 8085 and write the instructions in the instruction set of 8085 along with examples.
7. Write an assembly language code in GNUsim8085 to demonstrate immediate addressing.
8. Write an assembly language code in GNUsim8085 to demonstrate indirect addressing.
9. Write an assembly language code in GNUsim8085 to demonstrate indexed addressing.
10. Write an assembly language code in GNUsim8085 to implement data transfer instruction.

11. Write an assembly language code in GNUsim8085 to store numbers in reverse order in memory location.
12. Write an assembly language code in GNUsim8085 to implement arithmetic instruction.
13. Write an assembly language code in GNUsim8085 to perform addition of two 8 bit numbers.
14. Write an assembly language code in GNUsim8085 to perform addition of two 16 bit numbers.
15. Write an assembly language code in GNUsim8085 to find 1's & 2's complement of a 8 bit number.
16. Write an assembly language code in GNUsim8085 to add two 8 bit numbers stored in memory and also storing the carry.
17. Write an assembly language code in GNUsim8085 to find the factorial of a number.
18. Write an assembly language code in GNUsim8085 to implement logical instructions.
19. Write an assembly language code in GNUsim8085 to implement stack and branch instructions.
20. Write a program using in GNUsim8085 for Decimal addition and subtraction of two Numbers.
21. Write a program using in GNUsim8085 for Hexadecimal addition and subtraction of two Numbers.
22. Write a program using 8085 Microprocessor for addition and subtraction of two BCD numbers.
23. Write a program to perform multiplication of two 8 bit numbers using 8085.
24. Write a program to perform division of two 8 bit numbers using 8085.
25. Write a program to find the largest and smallest number in an array of data using 8085 instruction set.
26. Write a program to arrange an array of data in ascending and descending order.
27. Write an assembly language code in EMU8086 to store numbers in reverse order in memory location.
28. Write an assembly language code in EMU8086 to implement arithmetic instruction.
29. Write an assembly language code in EMU8086 to perform addition of two 8 bit numbers.
30. Write an assembly language code in EMU8086 to perform addition of two 16 bit numbers.

BACHELOR WITH COMPUTER APPLICATIONS AS MAJOR (CT-3)

6th SEMESTER

CAP622J3: COMPUTER APPLICATIONS _ PROBABILITY AND STATISTICS

CREDITS: THEORY (4) PRACTICAL (2)

COURSE LEARNING OUTCOMES:

1. *Learn the language and core concepts of probability theory.*
2. *Understand basic principles of statistical inference.*
3. *Become an informed consumer of statistical information and have a good knowledge of what expectation and variance mean and be able to compute them.*

PREREQUISITE: Fundamental Mathematics

UNIT 1: (15 LECTURES)

Introduction to Probability: Random experiment, sample space, trial, event. Simple probability, Compound Probability, mutually exclusive events, Addition theorem, independent events, Multiplication theorem, Dependent events, Conditional probability, Bayes' Theorem, Partitions and Total probability law. Exploring Univariate Data: Types of data, Mean, Mode and Median.

UNIT 2: (15 LECTURES)

Standard Deviation and Variance, Range and Finding Outliers. Counting, Random variables, probability mass function, probability density function. distributions, quantiles, mean-variance, Joint distributions, covariance, correlation, independence, and Central limit theorem.

Discrete Distributions, Random Variables, Binomial Distributions, Geometric Distributions Continuous Distributions, Density Curves, The Normal Distribution

UNIT 3: (15 LECTURES)

Multivariate Data, Scatter Plots, Correlation, The Least Squares Regression Line, Residuals, Non-Linear Models, Relations in Categorical Data, Margins of Error and Estimates, Confidence Interval for a Proportion, Confidence Interval for the Difference of Two Proportions, Confidence Interval for a Mean, Confidence Interval for the Difference of Two Means.

UNIT 4: (15 LECTURES)

Tests of Significance, Inference for the Mean of a Population, Sample Proportions, Inference for a Population Proportion, Comparing Two Means, Comparing Two Proportions, Goodness of Fit Test, and Two-way Tables. Simple correlation (Pearson's correlation coefficient), Simple linear regression, Prediction, error in prediction, principle of least square.

TUTORIALS:

1. Two dice are rolled. Find the sample space for this experiment.
2. What is the probability of drawing a red card from a standard deck of 52 cards?
3. If two events are mutually exclusive, and the probability of either event A or event B occurring is 0.4, what is P(A) + P(B)?
4. If two events are independent, and the probability of event A is 0.3 and the probability of event B is 0.6, what is P(A and B)?
5. If A and B are mutually exclusive events, and P(A) = 0.2, what is P(B)?
6. Calculate the conditional probability of event A given that event B has occurred if P(A) = 0.4 and P(B) = 0.3.
7. Using Bayes' Theorem, find the probability of event A if P(B/A) = 0.6, P(A) = 0.4, and P(B)= 0.5.
8. Define a random variable and provide an example of one.
9. If X is a discrete random variable with the following probability mass function: P(X = 1) = 0.2, P(X = 2) = 0.4, and P(X = 3) = 0.4, find E(X), the expected value of X.
10. In a deck of cards, what is the probability of drawing a face card (jack, queen, or king)?
11. If you roll a fair six-sided die three times, what is the probability of getting three 6s in a row?
12. If there are 25 students in a class, and 5 of them wear glasses, what is the probability that a randomly selected student does not wear glasses?
13. If a bag contains 8 white balls and 6 black balls, what is the probability of drawing a white ball and then a black ball (without replacement)?
14. A box contains 12 chocolates, of which 3 are dark chocolate. What is the probability of randomly selecting a dark chocolate?
15. If you have a deck of cards and draw 2 cards with replacement, what is the probability that both cards are aces?

16. If you flip a coin three times, what is the probability of getting exactly two heads?
17. If a jar contains 30 red marbles and 20 blue marbles, what is the probability of drawing a red marble on the first try and a blue marble on the second try (without replacement)?
18. If you roll a fair six-sided die four times, what is the probability of getting at least one 1?
19. Imagine you are flipping a fair coin. How can you use probability distributions to represent the outcomes of this coin toss experiment?
20. Suppose you have a standard six-sided die. What is the probability mass function for this die, and what is the probability of rolling a 3?
21. You are conducting a survey where each respondent can either say "Yes" or "No" to a question. If you expect 70% of people to say "Yes," can you use the binomial distribution to calculate the probability of getting exactly 4 "Yes" responses out of 10 respondents?
22. You are modeling the outcome of a single basketball free throw attempt, where a player either makes it (success) or misses it (failure). Is this scenario best represented by a Bernoulli distribution? Why or why not?
23. Imagine you are tracking the number of customer arrivals at a small coffee shop per hour. Can you explain when and why you might use a Poisson distribution to model this situation?
24. Consider the heights of adult males in a population. Why is the normal distribution often used to describe this data, and what are the defining characteristics of the normal distribution?
25. If a student's test score has a z-score of -1.5, what does this mean in terms of their performance compared to the rest of the students? How can you use percentiles to describe their ranking within the class?
26. Suppose you have a population of 100 test scores with a mean of 75 and a standard deviation of
10. If you take random samples of 30 test scores each and calculate the sample means, what is the expected mean of the sample means, and what is the standard error of the sample means? You're conducting a survey to estimate the average income of a population.
27. From a sample of 50 individuals, you find a sample mean income of 800000 and a sample standard deviation of 70000. Calculate a 95% confidence interval for the population mean income.
28. In a chi-square goodness-of-fit test, you expect a uniform distribution of colors in a bag of marbles, but you observe the following counts: Red (25), Blue (30), Green (20), and Yellow (25). Calculate the chi-square test statistic and determine if the observed distribution significantly differs from the expected uniform distribution at a 5% significance level
29. Suppose you have a dataset of 50 test scores, and the scores are normally distributed with an unknown mean (μ) and a known standard deviation (σ) of 15. If the maximum likelihood estimation gives you a mean of 65, what is the likelihood function for this dataset?
30. In a survey of 200 people, 120 said they prefer tea over coffee. Calculate a 95% confidence interval for the proportion of people in the entire population who prefer tea.

TEXTBOOK:

1. Probability and Statistics in Engineering (4th Edition) - W. Hines, D. Montgomery, D. Goldsman, C. Borror- Wiley Publication.

2. Introduction to Probability and Statistics for Engineers and Scientists (3rd Edition) - Sheldon M. Ross, Elsevier Academic Press.

REFERENCES:

1. Mood A.M. Graybill F.A and Boes D.C. (1974): Introduction to the Theory of Statistics. McGraw Hill.
2. Snedecor G.W and Cochran W.G. (1967); Statistical Methods. Lowa State University Press.
3. Cooke, Cramer and Clarke (1996): Basis Statistical Computing, Chapman and Hall. 4. David S. (1996): Elementary Probability, Oxford House.
4. Meyer P.L (1970): Introductory Probability and Statistical application, Addison Wesley.

BACHELOR WITH APPLIED COMPUTING AS MINOR
SIXTH SEMESTER

(FOR STUDENTS WITH MAJOR IN COMPUTER APPLICATIONS / INFORMATION TECHNOLOGY)

ACP622N: APPLIED COMPUTING _ ARTIFICIAL INTELLIGENCE

CREDITS: THEORY (3) PRACTICAL (1)

MAX. MARKS: 100 MIN. MARKS: 36

COURSE LEARNING OUTCOMES:

1. *Study the concepts of Artificial Intelligence.*
2. *Learn the methods of solving problems using Artificial Intelligence.*
3. *Learn the knowledge representation techniques, reasoning techniques and planning.*

THEORY (3 CREDITS)

PREREQUISITE: Data Structures and Analysis of Algorithm

UNIT 1: AI AND TURING TEST: (15 LECTURES)

Introduction to Artificial Intelligence, Background and Applications, Turing Test, Rational Agent approaches to AI, Introduction to Intelligent Agents, structure, behaviour and environment.

UNIT 2: SEARCHING PROBLEM SPACE: (15 LECTURES)

Problem Characteristics, Production Systems, Informed vs Uninformed Strategies, Hill Climbing and its Variations, Heuristics Search Techniques: Best First Search, A* algorithm.
The Wumpus World Environment, Acting and reasoning in the Wumpus world, Representation, Reasoning, Logic, Prepositional Logic and First-Order Logic, Rules of Inferencing: Resolution Principle.

UNIT 3: UNCERTAIN KNOWLEDGE AND REASONING: (15 LECTURES)

Acting under Uncertainty, handling uncertain knowledge, Uncertainty and rational decisions, Basic Probability Notation, Prior probability, Conditional probability, joint probability distribution, Bayes' Rule and Its Use, Applying Bayes' rule: The simple case, Normalization, Dampster's Shafer's Theorem.

TUTORIAL (1 CREDIT):

1. Given a real life situation
   - identify the characteristics of the environment
   - identify the percepts available to the agent
   - identify the actions that the agent can execute
   - suggest the performance measures to evaluate the agent
   - recommend the architecture of the desired intelligent agent
2. Undertake a comparative study on popular programming languages used in AI.
3. Translate commonly used English sentences into their equivalent logical expressions.
4. Given a problem description, formulate it in terms of a state space search problem. Analyze the given problem and identify the most suitable search strategy for the problem. Make an analysis of the properties of proposed algorithms in terms of
   - time complexity
   - space complexity
   - optimality
5. Write a Program to Implement Breadth First Search.
6. Write a Program to Implement Depth First Search.
7. Write a program to implement Hill Climbing Algorithm
8. Write a program to implement A* Algorithm.
9. Write a program to implement Tic-Tac-Toe game.
10. Write a Program to Implement 8-Puzzle problem.
11. Write a Program to Implement Water-Jug problem.
12. Write a Program to Implement Alpha-Beta Pruning using Python.
13. Write a Program to Implement 8-Queens Problem.
14. Build a Fuzzy inference system for the Tipping Problem.
    Given two sets of numbers between 0 and 5 (where 0 is for very poor, and 5 for excellent) that respectively represent quality of service and quality of food at restaurant, what should tip be?
15. Solve 2-input 1-output project risk prediction problem using Mamdani Inference. Make necessary assumptions.

TEXT BOOK:

1. DAN.W. Patterson, Introduction to A.I and Expert Systems - PHI, 2007.

REFERENCES:

1. Russell & Norvig. Artificial Intelligence-A Modern Approach, LPE, Pearson Prentice Hall, 2nd edition. 2005.
2. Ivan Bratko,"Prolog Programming for Artificial Intelligence", 3 rd Edition, Pearson Education.
3. Rich & Knight, Artificial Intelligence - Tata McGraw Hill. 2nd edition, 1991.
4. W.F. Clocksin and Mellish, Programming in PROLOG, Narosa Publishing House, 3rd edition, 2001.

BACHELOR WITH COMPUTER APPLICATIONS AS MINOR (CT-1)
6th SEMESTER

CAP622N: COMPUTER APPLICATIONS _ PYTHON PROGRAMMING

CREDITS: THEORY (3) PRACTICAL (1)

LEARNING OUTCOMES:

After completing this course, the learner shall be able to:

1. *Understand the fundamentals of programming through Python*
2. *Transform a solution from a subjective world into an objective world.*
3. *Work with and manipulate different basic data structures available in Python*
4. *Use the List comprehensions and generators in their programs*
5. *Apply basic object-oriented concepts to design classes and objects in Python*
6. *Use concepts of Inheritance and Polymorphism in their programs*
7. *Perform basic file operations for text and CSV files*
8. *Use existing inbuilt Python modules*
9. *Develop custom modules*
10. *Use basic functionalities provided by packages Numpy and Pandas*
11. *Develop visualizations using different plotting functions available in matplotlib*

UNIT-I: (15 LECTURES)

Origins and History of Python, Structure of a Python Program, Interpreter shell, Indentation, Comments, Identifiers and keywords, Literals, Basic operators (Arithmetic operator, Relational operator, Logical or Boolean operator, Assignment Operator, Bit wise operator)

Entering Expressions into the Interactive Shell, The Integer, Floating-Point, and String Data Types, String Concatenation and Replication, Storing Values in Variables. Creating Python Programs: Input and Output Statements, Control statements: -branching, looping, range function, Exit function, break, continue and pass.

UNIT: II (15 LECTURES)

Functions: Defining a Function, Calling a Function, def Statements with Parameters, Formal and Actual Arguments, Positional Arguments, Keyword Arguments, Default Arguments, Variable Length Arguments, Return Values and return Statements, The None Value, Local and Global Scope, The global Statement.

The List Data Type: Creating Lists, Basic List Operations, Indexing and Slicing in Lists, Built-In Functions Used on Lists, List Methods. List comprehension.

UNIT: III (15 LECTURES)

The Dictionary Data Type: Creating Dictionary, Accessing and Modifying key: value Pairs in Dictionaries, Built-In Functions used on Dictionaries, Dictionary Methods

Tuples and Sets, Creating Tuples, Basic Tuple Operations, Indexing and Slicing in Tuples, Built-In Functions Used on Tuples, Set Methods, Traversing of Sets

Strings, Creating and Storing Strings, Basic String Operations, Accessing Characters in String by Index Number, String Slicing and Joining, String Methods, Formatting Strings.

Files: Types of Files, Creating and Reading Text Data

TEXT BOOKS

1. Gowrishankar S, Veena A, “Introduction to Python Programming”, 1st Edition, CRC Press/Taylor & Francis, 2018. ISBN-13: 978-0815394372
2. Downey, A.B., (2015), Think Python How to think like a Computer Scientist, 3rd edition.
3. Taneja, S. & Kumar, N., (2017), Python Programming- A Modular Approach. Pearson Education.

REFERENCE BOOKS

1. Jake VanderPlas, “Python Data Science Handbook: Essential Tools for Working with Data”, 1st Edition, O'Reilly Media, 2016. ISBN-13: 978-1491912058
2. Aurelien Geron, “Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems”, 2nd Edition, O’Reilly Media, 2019. ISBN – 13: 978-9352139057.
3. Wesley J Chun, “Core Python Applications Programming”, 3rd Edition, Pearson Education India, 2015. ISBN-13: 978-9332555365

4. Miguel Grinberg, “Flask Web Development: Developing Web Applications with Python”, 2nd Edition, O'Reilly Media, 2018. ISBN-13: 978-1491991732.
5. Brown, M. C. (2001). The Complete Reference: Python, McGraw Hill Education.
6. Dromey, R. G. (2006), How to Solve it by Computer, Pearson Education.
7. Guttag, J. V. (2016), Introduction to computation and programming using Python. MIT Press.
8. Liang, Y.D. (2013), Introduction to programming using Python. Pearson Education.

LAB WORK: PYTHON PROGRAMMING

1. Install and configure Python IDE
2. Write simple Python program to display message on screen
3. Write a program to demonstrate different number data types & arithmetic operations in Python.
4. Write a Python program to demonstrate use of conditional statements:
5. Write Python program to demonstrate use of looping statements
6. Demonstrate the following control transfer statements in Python with suitable examples.
    i) break
    ii) continue
    iii) pass
7. Write a python program to demonstrate the working of following functions in Python.
    i) id( )
    ii) type( )
    iii) range( )
8. Write Python programs to demonstrate the following:
    i) input( )
    ii) print( )
    iii) ‘sep’ attribute
    iv) ‘end’ attribute
9. Write a program to create a menu driven program with the following options using functions:
    Addition 2. Subtraction 3. Multiplication 4. Division
10. Write Python program to perform following operations on Lists:
    a) Create list
    b) Access list
    c) Update list (Add item, Remove item)
    d) Delete list
11. Write Python program to perform following operations on Tuples:
    a) Create Tuple
    b) Access Tuple
    c) Update Tuple
    d) Delete Tuple
12. Write Python program to perform following operations on Sets:
    a) Create Set
    b) Access Set elements
    c) Update Set
    d) Delete Set
13. Write Python program to perform following operations on Dictionaries:
    a) Create Dictionary
    b) Access Dictionary elements
    c) Update Dictionary
    d) Delete Set
    e) Looping through Dictionary
14. Write a Python program to demonstrate slicing operations on lists and strings.
15. Write a Python program to perform read and write operations on a file.