UNIT-I

1. List and explain different arithmetic operators supported by Python. Discuss about their precedence and associativity.
2. Python has developed as an open source project. Justify this statement
3. Give a note on each of the below Python language constructs: quotes (single, double and triple) (ii) multiline statements (iii) indentation
4. What are different applications of Python? Give examples.
5. Explain in detail about variables and identifiers?
6. What is a literal? Explain different types of literals with an examples
7. What is a control structure? List and explain fundamental forms of control in programming.
8. Define Computational problem. Explain the process of Computational problem solving in detail.
9. What is a literal? Explain integer and floating point literals with examples.
10. Explain selection control statements in python with examples.
11. Write python code that use a while loop to add up all the even numbers between 100 and 200 inclusive.
12. Explain the Features of Python Programming Language.
13. What is Data type? What are the Different data types available in Python? Explain Briefly.
14. What are the different Selection Control Statements in Python? Explain with example Programs.
15. Explain the process of computational problem solving?
16. Discuss in brief about operators in python with an example?
17. Briefly discuss about the looping techniques in Python with suitable examples?
18. Write a Python program to find number of days in a month.
19. Write a Python program to find maximum number among three numbers.
20. Write a Python program to find whether number is prime or not.
21. Write a Python program to find first n number of Fibonacci series.
22. Write a Python program to find whether given number is Armstrong or not.
23. Write a Python program to find reverse of the given number.
24. Write a Python program to convert temperature from Fahrenheit to Celsius and vice versa.
25. Write a Python program to calculate age in seconds.
26. Python programs on any pattern.

UNIT-II

1. Explain the Python Dictionary Comprehension with examples.
2. Write a Python program that calculates number of seconds in a day.
3. Explain the List Accessing Methods and List Comprehension.
4. Write a Python program to read a word and print the number of letters, vowels and percentage of vowels in the word using a dictionary.
5. What is a list and differentiate the lists in Python?
6. Differentiate calling value-returning function and calling non-value-returning function?
7. Explain about parameter passing mechanisms?
8. What is a list? Explain common list operations with examples.
9. How can you iterate lists in python? Explain iterating over lists Using

   (i) for loop ii) for loop with range function iii) while loop
10. Briefly discuss the following with simple examples.

    (i) Positional Arguments

    (ii) Mutable & Immutable arguments

    (iii) keyword arguments

    (iv) Default arguments

    (v) Variable length arguments
11. Illustrate Iterating Over Lists in Python.
12. Explain the dictionary type in python?
13. Explain about i) Tuples ii) Set
14. Briefly discuss about the program routines?
15. Explain the following operations with suitable examples.

    (i) Append()

    (ii) insert()

    (iii) extend()

    (iv) remove() & pop()
16. Write a Python Program to find the cubes of numbers from 1 to 20 using list comprehension.
17. What is recursion? Write a Python program to find factorial of a given number using recursion.
18. Write a Python Program to read elements into the list and also find maximum element in the list.
19. Write a Python program to encrypt and decrypt the password using tuple.
20. Write a python program to display temperature in a day using dictionary.
21. Write a python program to find average of three numbers using function.

UNIT-III

1. What are the two ways of importing a module? Which one is more beneficial? Explain.
2. Discuss exceptions with arguments in Python with suitable example.
3. Why to use modules? How to structure a program?
4. What is a text file? How to use text file? Explain.
5. Develop a modular design of the Calendar program.
6. What is a module? Explain about Module specification?
7. What is a text file? Explain about how to use text files?
8. Explain in detail about exception handling mechanisms.
9. What are python module and name spaces? How can you import modules? Explain with example.
10. Compare and contrast local, global and built-in namespaces in python.
11. What is a text file? Explain the fundamental file operations with syntax and examples
12. Write a python segment that opens and reads a text file and displays how many lines of text are in the file.
13. Briefly Explain Python Modules.
14. How to handle Exceptions in python? Discuss in brief with example
15. Explain in detail about String Processing in python.
16. Define module? List different python modules and explain any one.
17. Explain in detail about reading and writing text files in python.
18. Write a Python Program to implement calendar month using module design.
19. Write a Python Program to Copying Data From One File To Other File:
20. Write a Python Program To Copying Even Numbers To Even.Txt and Odd Numbers To Odd.Txt.
21. Explain string processing methods with suitable examples.
22. Explain the occurrences of the following standard exceptions with suitable examples.
    - (i) Import Error
    - (ii) Index Error
    - (iii) Name Error
    - (iv) Type Error
    - (v) Value Error
23. How is raised exception propagated with in functions? Explain with suitable figure.
24. Define error and exception. Distinguish between these two features

UNIT-IV

1. How to declare a constructor method in Python? Explain.
2. Explain the concept of creating classes in Python with examples.
3. List the features and explain about different Object Oriented features supported by Python.
4. What is an object? Explain in detail about object references?
5. What is encapsulation? Explain with suitable Example?
6. Explain how memory allocation, de-allocation and garbage collection in python.
7. What is Object oriented programming? List and explain three fundamental features of object oriented programming?
8. Describe the following
    (i) Encapsulation
    (ii) Inheritance
    (iii) Polymorphism
9. Explain classes and objects in python.
10. Define software object. Explain object references.
11. Compare deep copy and shallow copy with an example.
12. How to use self-parameter in python? Explain with suitable example.
13. What is Inheritance? Explain different types of inheritance in python.
14. What is Polymorphism? Explain with suitable example.
15. What is name mangling? Explain with suitable example.
16. Explain the following concepts with suitable example.
    (i) Class Variable
    (ii) Instance variable
    (iii) Local Variable
    (iv) Global variable
    (v) __init__() method

UNIT-V

1. Explain single linked list representation
2. Explain Stacks-implementation using python list with suitable program.
3. How to implement stacks using Python list and linked list?
4. How to implement queues using Python list and linked list?
5. Define abstract data type. Explain with example.
6. Write python code to perform prepending, traversing, searching and removing in the linked list.
7. Write a python program to implement stack operations.
8. What is ADT? Explain about Single-Linked List with example Program.
9. Define Queue ADT and implement queue using python linked list.
10. Define Stack ADT and implement stack using a python list.
11. Write a python program to implement Bag ADT using single linked list.
12. What is ADT? Explain its advantages in python.
13. What is Iterator? How to iterate python objects? Explain iter() and next() methods in python with suitable example.
14. Explain the following linked list operations with suitable diagrams.
    - (i) Prepending
    - (ii) Traversing
    - (iii) Searching
    - (iv) Removing