PARVATHANENI BRAHMAYYA
SIDDHARTHA COLLEGE OF ARTS & SCIENCE
Autonomous
Siddhartha Nagar, Vijayawada–520010
Re-accredited at 'A+' by the NAAC

23AIMAL121: Python Programming **Year of introduction: 2023-24**
Offered to : B.Sc. Honours (Artificial Intelligence) **Course Type : Major 3(TH)**
Teaching Periods: 60 **Semester - II** **No. of Credits:3**

Course OBJECTIVE: After taking the course, students will be able to use Python program a Scripting language and Exposure of various problems soling approaches of computer

COURSE OUTCOME NO	Upon successful completion of this course, the student will be able to	PROGRAM OUTCOME NO
CO1	Learn about concepts of programming and python	**PO1, PO2**
CO2	Understand the Decision making and looping controls available in Python Programs	**PO2, PO3**
CO3	Determine the process of using functions and modules	**PO3, PO4**
CO4	Implement the Data structures using Lists, Tuple, Dictionaries	**PO4, PO3**
CO5	Interpret the OOPs concept in Python.	**PO5, PO3**

CO-PO MAPPING MATRIX

	CO - PO	PO1	PO2	PO3	PO4	PO5	PO6
23AIMAL121	**CO1**	H	M				
	CO2		H	M			
	CO3			H	L		
	CO4			L	H		
	CO5			H		L	

UNIT – I: Introduction, Data types, Operators: **12 Periods**

Introduction to Programming: Languages, Generations, Programming Paradigms, Debugging and Testing Approaches.
Python : Introduction, History of Python, Features of Python, Writing and executing python programs, constants, variables, reserved words, input operation, indentation,
Data types in python: integer, string, Boolean, Operators and expressions.

UNIT – II: Decision making and looping **13 Periods**

Control Flow: **Selection or conditional branching** : if, if-else, if-elif- else, nested if.
Loop or iterative statements: for, while, break, continue, pass. Example programs on control flow.

UNIT – III: Functions and Modules: **12 Periods**

Defining Functions, Calling Functions, Passing Arguments, Keyword Arguments, Default Arguments, Variable-length arguments, Lambda or Anonymous Functions, Global and Local Variables.
Modules: Creating modules, import statement, from import statement.

UNIT – IV: Data Structures: **13 Periods**

Lists : Creation, accessing values, updating values, list operations, list methods.
Tuple: tuple creation, accessing values, deleting values, tuple operations. **Sets:** creation, set operations.
Dictionaries: creation, accessing values, adding , modifying , deleting items, built-in dictionary methods.
UNIT – IV: Object Oriented Programming in Python: **10 Periods**

Introduction to OOP, Features of OOP, Merits and Demerits, Classes and Objects, Class method and self-Argument, Public and Private, the init method(constructor), Inheritance, polymorphism and Method Overriding.
Error and Exception handling: Handling Exception using try-except block, Raising Exceptions, User Defined Exceptions.

TEXT BOOKS

Python Programming: Using Problem Solving approach, Reema Thareja, Oxford University Press 2017
REFERENCE BOOKS:

PYTHON PROGRAMMING A Modern Approach, Vamsi Kurama, Pearson Publications, 2017
Recommended Co – Curricular Activities:

(Co-curricular activities shall not promote copying from textbook or from others work and shall encourage self/independent and group learning)

A Measurable

a. Assignments (in writing and doing forms on the aspects of syllabus content and outside the syllabus content. Shall be individual and challenging).
b. Student seminars (on topics of the syllabus and related aspects (individual activity))
c. Quiz (on topics where the content can be compiled by smaller aspects and data (Individuals or groups as teams))
d. Study projects (by very small groups of students on selected local real- time problems pertaining to syllabus or related areas. The individual participation and contribution of students shall be ensured (team activity))

B General

a. Group Discussion
b. Others

RECOMMENDED CONTINUOUS ASSESSMENT METHODS:

Some of the following suggested assessment methodologies could be adopted;

1. The oral and written examinations (Scheduled and surprise tests),
2. Closed-book and open-book tests,
3. Programming exercises,
4. Practical assignments and laboratory reports,
5. Observation of practical skills,
6. Individual and group project reports.
7. Efficient delivery using seminar presentations,
8. Viva voce interviews.
9. Computerized adaptive testing, literature surveys and evaluations,
10. Peers and self-assessment, outputs form individual and collaborative work.

@@@@

PARVATHANENI BRAHMAYYA
SIDDHARTHA COLLEGE OF ARTS & SCIENCE
Autonomous
Siddhartha Nagar, Vijayawada–520010
Re-accredited at 'A+' by the NAAC

MODEL PAPER

23AIMAL121:Python Programming **B.Sc. (Honours) Artificial Intelligence**

SEMESTER: II

TIME: 3 Hrs. **Max.Marks : 70M**

Note to paper setter: In Section A & Section B, for each question one sub question (A) must be a program meant for logical testing and another sub question (B) is meant for descriptive / Logical.

SECTION A

ANSWER THE FOLLOWING QUESTIONS. **5 X 4 = 20 Marks**

1. a) Write different types of programming paradigms. (CO1, L1)

   OR

   b) Develop a python code to print sum and average marks of three subjects of a student. (CO1, L1)
2. a) Write a python program to find the factorial of a given number. (CO2, L1)

   OR

   b) Explain if-elif-else statement in python with example (CO2, L2)
3. a) Develop a python program for Lambda function. (CO3, L3)

   OR

   b) Explain about global and local variables in python. (CO3, L2)
4. a) Develop a python program to insert elements into a list, remove elements from a list and sort elements of the list. (CO4, L3)

   OR

   b) Compare lists and tuples in python. (CO4, L3)
5. a) Develop a python program to demonstrate Inheritance. (CO5, L3)

   OR

   b) Explain about classes and objects in python. (CO5, L2)

SECTION B

ANSWER THE FOLLOWING QUESTIONS . **5 X 10 = 50 Marks**

6. a) List and explain data types in python. (CO1, L2)

   OR

   b) Write names of operators in python and explain them with examples. (CO1, L2)
7. a) Develop a python program to find whether the number is palindrome or not. (CO2, L3)

   OR

   b) Explain loops in Python with examples. (CO2, L2)
8. a) Explain different categories of arguments used in functions in python. (CO3, L3)

   OR

   b) Develop a python program to create a module and import it in another program. (CO3, L2)
9. a) Develop a python program to create a dictionary and add, modify, delete values in the dictionary and print them. (CO4, L2)

   OR

   b) Define tuple. Explain about tuples with examples. (CO4, L3)
10. a) Develop a python program to demonstrate exception handling. (CO5, L3)

    OR

    b) Define method overriding. Explain with an example. (CO5, L2)

@@@@

PARVATHANENI BRAHMAYYA
SIDDHARTHA COLLEGE OF ARTS & SCIENCE
Autonomous
Siddhartha Nagar, Vijayawada–520010
Re-accredited at 'A+' by the NAAC

23AIMAP121: Python Programming Lab **Year of Introduction: 2023-24**
Offered to : B.Sc. Honours (Artificial Intelligence) **Course Type : Major 3(P)**
Teaching Periods: 30 **Semester - II** **No. of Credits:1**

COURSE OUTCOME NO	Upon successful completion of this course, the student will be able to	PROGRAM OUTCOME NO
CO1	Write, Test and Debug Python Programs	**PO1**
CO2	Implement Conditionals and Loops for Python Programs	**PO1,PO2,PO3**
CO3	Organize code into modules for better code organization and reusability.	**PO2,PO3**
CO4	Implement functions and represent Compound data using Lists, Tuples and Dictionaries	**PO3,PO4**
CO5	Implement OOP concepts and write applications in python.	**PO5,PO3**

CO-PO MAPPING MATRIX

	CO - PO	PO1	PO2	PO3	PO4	PO5	PO6
23AIMAP121	**CO1**	H					
	CO2	L	H	M			
	CO3		H	M			
	CO4			L	H		
	CO5			H		L	

List of Experiments:

1. Python Program to Find the Square Root
2. Python Program to Swap Two Variables
3. Python Program to Generate a Random Number
4. Python Program to Check if a Number is Odd or Even
5. Python Program to Find the Largest Among Four Numbers
6. Python Program to Check Prime Number
7. Python Program to Check Whether a number is Palindrome or Not
8. Python Program to Display the multiplication Table
9. Python Program to Print the Fibonacci sequence

10. Python Program to Check Armstrong Number
11. Python Program to Find the Sum of Natural Numbers
12. Python Program to Find Factorial of Number Using Recursion
13. Python Program to check given number is prime or not using functions.
14. Python Program to demonstrate usage of keyword, default and variable length arguments.
15. Python Program for lambda functions.
16. Python Program to create module and import it.
17. Python Program to create a list and perform operations on its contents.
18. Python Program to perform operations on tuples.
19. Python Program to create a dictionary and print its content.
20. Python program to perform operations on sets.
21. Python Program for inheritance.
22. Python Program for method overriding.
23. Python Program for exception handling.
24. Python Program to demonstrate exception handling.
25. Python Program to demonstrate user defined exception.

Question Paper Pattern for Practical Courses

23AIMAP121: Python Programming Lab **Offered to : B.Sc. Honours (Artificial Intelligence)**
SEMESTER: II **Max. Marks : 50 (CIA: 15 + SEE: 35)**

Model Paper : Practicals

Time: 3 Hrs **Max. Marks : 35**

Section - A

1.	Experiment 1	15 M
2.	Experiment 2	10 M

Section – B

Viva Voce	10 M
CONTINUOUS ASSESMENT (Internal)	**15 M**
Total	50 M

PARVATHANENI BRAHMAYYA
SIDDHARTHA COLLEGE OF ARTS & SCIENCE
Autonomous
Siddhartha Nagar, Vijayawada–520010
Re-accredited at 'A+' by the NAAC

23DSMAL121: Python Programming

Programme: B. Sc. Hons. (Data Science) **Semester: II**

Teaching Periods: 60 **No. of Credits:3**

Course OBJECTIVE: After taking the course, students will be able to use Python program a Scripting language and Exposure of various problems soling approaches of computer

COURSE OUTCOME NO	Upon successful completion of this course, the student will be able to	PROGRAM OUTCOME NO
CO1	Learn about concepts of programming and python	**PO1, PO2**
CO2	Understand the Decision making and looping controls available in Python Programs	**PO2, PO3**
CO3	Determine the process of using functions and modules	**PO3, PO4**
CO4	Implement the Data structures using Lists, Tuple, Dictionaries	**PO4, PO3**
CO5	Interpret the OOPs concept in Python.	**PO5, PO3**

CO-PO MAPPING MATRIX

	CO – PO	PO1	PO2	PO3	PO4	PO5	PO6
23DSMAL121	**CO1**	H	M				
	CO2		H	M			
	CO3			H	L		
	CO4			L	H		
	CO5			H		L	

UNIT – I: Introduction, Data types, Operators: **12 Periods**

Introduction to Programming: Languages , Generations, Programming Paradigms, Debugging and Testing Approaches.

Python : Introduction, History of Python, Features of Python, Writing and executing python programs, constants, variables, reserved words, input operation, indentation,

Data types in python: integer, string, Boolean, Operators and expressions.

UNIT – II: Decision making and looping **13 Periods**

Control Flow: **Selection or conditional branching** : if, if-else, if-elif- else, nested if.
Loop or iterative statements: for, while, break, continue, pass. Example programs on control flow.

UNIT – III: Functions and Modules: **12 Periods**

Defining Functions, Calling Functions, Passing Arguments, Keyword Arguments, Default Arguments, Variable-length arguments, Lambda or Anonymous Functions, Global and Local Variables.
Modules: Creating modules, import statement, from import statement.

UNIT – IV: Data Structures: **13 Periods**

Lists : Creation, accessing values, updating values, list operations, list methods.
Tuple: tuple creation, accessing values, deleting values, tuple operations. **Sets:** creation, set operations. **Dictionaries:** creation, accessing values, adding , modifying , deleting items, built-in dictionary methods.

UNIT – IV: Object Oriented Programming in Python: **10 Periods**

Introduction to OOP, Features of OOP, Merits and Demerits, Classes and Objects, Class method and self Argument, Public and Private , the init method(constructor), Inheritance, polymorphism and Method Overriding. **Error and Exception handling**: Handling Exception using try-except block, Raising Exceptions, User Defined Exceptions.

TEXT BOOKS

Python Programming: Using Problem Solving approach, Reema Thareja, Oxford University Press 2017

REFERENCE BOOKS:
PYTHON PROGRAMMING A Modern Approach, Vamsi Kurama, Pearson Publications, 2017

Recommended Co – Curricular Activities:

(Co-curricular activities shall not promote copying from textbook or from others work and shall encourage self/independent and group learning)

A. **Measurable**

- **a.** Assignments (in writing and doing forms on the aspects of syllabus content and outside the syllabus content. Shall be individual and challenging).
- **b.** Student seminars (on topics of the syllabus and related aspects (individual activity))
- **c.** Quiz (on topics where the content can be compiled by smaller aspects and data (Individuals or groups as teams))
- **d.** Study projects (by very small groups of students on selected local real- time problems pertaining to syllabus or related areas. The individual participation and contribution of students shall be ensured (team activity))

B. General

a. Group Discussion
b. Others

RECOMMENDED CONTINUOUS ASSESSMENT METHODS:

Some of the following suggested assessment methodologies could be adopted;

1. The oral and written examinations (Scheduled and surprise tests),
2. Closed-book and open-book tests,
3. Programming exercises,
4. Practical assignments and laboratory reports,
5. Observation of practical skills,
6. Individual and group project reports.
7. Efficient delivery using seminar presentations,
8. Viva voce interviews.
9. Computerized adaptive testing, literature surveys and evaluations,
10. Peers and self-assessment, outputs form individual and collaborative work.

@@@@

PARVATHANENI BRAHMAYYA
SIDDHARTHA COLLEGE OF ARTS & SCIENCE
Autonomous
Siddhartha Nagar, Vijayawada–520010
Re-accredited at 'A+' by the NAAC

MODEL PAPER

TITLE: Python Programming **COURSE CODE: 23DSMAL121**
SECTION: B.Sc. (Honours) Data Science **SEMESTER: II**
TIME: 3 Hrs. **MAX: 70M**

NOTE TO PAPER SETTER: IN SECTION A & SECTION B, FOR EACH QUESTION ONE SUB QUESTION (A) MUST BE A PROGRAM MEANT FOR LOGICAL TESTING AND ANOTHER SUB QUESTION (B) IS MEANT FOR DESCRIPTIVE / LOGICAL.

SECTION A

ANSWER THE FOLLOWING QUESTIONS. **5 X 4 = 20 Marks**

1. a) Write different types of programming paradigms. (CO1, L1)
   OR
   b) Develop a python code to print sum and average marks of three subjects of a student. (CO1, L1)
2. a) Write a python program to find the factorial of a given number. (CO2, L1)
   OR
   b) Explain if-elif-else statement in python with example (CO2, L2)
3. a) Develop a python program for Lambda function. (CO3, L3)
   OR
   b) Explain about global and local variables in python. (CO3, L2)
4. a) Develop a python program to insert elements into a list, remove elements from a list and sort elements of the list. (CO4, L3)
   OR
   b) Compare lists and tuples in python. (CO4, L3)
5. a) Develop a python program to demonstrate Inheritance. (CO5, L3)
   OR
   b) Explain about classes and objects in python. (CO5, L2)

SECTION B

ANSWER THE FOLLOWING QUESTIONS . **5 X 10 = 50 Marks**

6. a) List and explain data types in python. (CO1, L2)
   OR
   b) Write names of operators in python and explain them with examples. (CO1, L2)
7. a) Develop a python program to find whether the number is palindrome or not. (CO2, L3)
   OR
   b) Explain loops in Python with examples. (CO2, L2)
8. a) Explain different categories of arguments used in functions in python. (CO3, L3)
   OR
   b) Develop a python program to create a module and import it in another program. (CO3, L2)
9. a) Develop a python program to create a dictionary and add, modify, delete values in the dictionary and print them. (CO4, L2)
   OR
   b) Define tuple. Explain about tuples with examples. (CO4, L3)
10. a) Develop a python program to demonstrate exception handling. (CO5, L3)
    OR
    b) Define method overriding. Explain with an example. (CO5, L2)

@@@@

PARVATHANENI BRAHMAYYA
SIDDHARTHA COLLEGE OF ARTS & SCIENCE
Autonomous
Siddhartha Nagar, Vijayawada–520010
Re-accredited at 'A+' by the NAAC

23DSMAP121: Python Programming Lab

Programme: B. Sc. Honours (Data Science) **Semester: II**
Teaching Periods: 30 **No. of Credits: 1**

COURSE OUTCOME NO	Upon successful completion of this course, the student will be able to	PROGRAM OUTCOME NO
CO1	Write, Test and Debug Python Programs	**PO1**
CO2	Implement Conditionals and Loops for Python Programs	**PO1,PO2,PO3**
CO3	Organize code into modules for better code organization and reusability.	**PO2,PO3**
CO4	Use functions and represent Compound data using Lists, Tuples and Dictionaries	**PO3,PO4**
CO5	Implement OOP concepts and write applications in python.	**PO5,PO3**

CO-PO MAPPING MATRIX

	CO - PO	PO1	PO2	PO3	PO4	PO5	PO6
23DSMAP121	**CO1**	H					
	CO2	L	H	M			
	CO3		H	M			
	CO4			L	H		
	CO5			H		L	

List of Experiments:

1. Python Program to Find the Square Root
2. Python Program to Swap Two Variables
3. Python Program to Generate a Random Number
4. Python Program to Check if a Number is Odd or Even
5. Python Program to Find the Largest Among Four Numbers
6. Python Program to Check Prime Number
7. Python Program to Check Whether a number is Palindrome or Not
8. Python Program to Display the multiplication Table
9. Python Program to Print the Fibonacci sequence
10. Python Program to Check Armstrong Number
11. Python Program to Find the Sum of Natural Numbers
12. Python Program to Find Factorial of Number Using Recursion

13. Python Program to check given number is prime or not using functions.
14. Python Program to demonstrate usage of keyword, default and variable length arguments.
15. Python Program for lambda functions.
16. Python Program to create a module and import it.
17. Python Program to create a list and perform operations on its contents.
18. Python Program to perform operations on tuples.
19. Python Program to create a dictionary and print its content.
20. Python program to perform operations on sets.
21. Python Program for inheritance.
22. Python Program for method overriding.
23. Python Program for exception handling.

Question Paper Pattern for Practical Course

TITLE: Python Programming Lab **COURSE CODE: 23DSMAP121**

SECTION: B.Sc. (Honours) Data Science **SEMESTER: II**

Max. Marks : 50 (CIA: 15 + SEE: 35) **Hrs/Week: 2**

Time: 3 Hrs **Max. Marks : 35**

Section - A

1. Experiment 1	15 M
2. Experiment 2	10 M

Section – B

Viva Voce	10 M
CONTINUOUS ASSESMENT(Internal)	**15 M**
Total	50 M

PARVATHANENI BRAHMAYYA
SIDDHARTHA COLLEGE OF ARTS & SCIENCE
Autonomous
Siddhartha Nagar, Vijayawada–520010
Re-accredited at 'A+' by the NAAC

23DAMAL121: Python Programming

Offered to : B.Sc. Honours (Data Analytics) **Semester : II**

Teaching Periods: 60 **No. of Credits:3**

Course OBJECTIVE: After taking the course, students will be able to use Python program a Scripting language and Exposure of various problems soling approaches of computer

COURSE OUTCOME NO	Upon successful completion of this course, the student will be able to	PROGRAM OUTCOME NO
CO1	Learn about concepts of programming and python	**PO1, PO2**
CO2	Understand the Decision making and looping controls available in Python Programs	**PO2, PO3**
CO3	Determine the process of using functions and modules	**PO3, PO4**
CO4	Implement the Data structures using Lists, Tuple, Dictionaries	**PO4, PO3**
CO5	Interpret the OOPs concept in Python.	**PO5, PO3**

CO-PO MAPPING MATRIX

	CO - PO	PO1	PO2	PO3	PO4	PO5	PO6
23DAMAL121	**CO1**	H	M				
	CO2		H	M			
	CO3			H	L		
	CO4			L	H		
	CO5			H		L	

UNIT – I: Introduction, Data types, Operators: **12 Periods**

Introduction to Programming: Languages , Generations, Programming Paradigms, Debugging and Testing Approaches.

Python : Introduction, History of Python, Features of Python, Writing and executing python programs, constants, variables, reserved words, input operation, indentation,

Data types in python: integer, string, Boolean, Operators and expressions.

UNIT – II: Decision making and looping **13 Periods**

Control Flow: **Selection or conditional branching** : if, if-else, if-elif- else, nested if.
Loop or iterative statements: for, while, break, continue, pass. Example programs on control flow.

UNIT – III: Functions and Modules: **12 Periods**

Defining Functions, Calling Functions, Passing Arguments, Keyword Arguments, Default Arguments, Variable-length arguments, Lambda or Anonymous Functions, Global and Local Variables.
Modules: Creating modules, import statement, from import statement.

UNIT – IV: Data Structures: **13 Periods**

Lists : Creation, accessing values, updating values, list operations, list methods.
Tuple: tuple creation, accessing values, deleting values, tuple operations. **Sets:** creation, set operations. **Dictionaries:** creation, accessing values, adding , modifying , deleting items, built-in dictionary methods.

UNIT – IV: Object Oriented Programming in Python: **10 Periods**

Introduction to OOP, Features of OOP, Merits and Demerits, Classes and Objects, Class method and self-Argument, Public and Private , the init method(constructor), Inheritance, polymorphism and Method Overriding. **Error and Exception handling**: Handling Exception using try-except block, Raising Exceptions, User Defined Exceptions.

TEXT BOOKS

Python Programming: Using Problem Solving approach, Reema Thareja, Oxford University Press 2017

REFERENCE BOOKS:

PYTHON PROGRAMMING A Modern Approach, Vamsi Kurama, Pearson Publications, 2017

Recommended Co – Curricular Activities:

(Co-curricular activities shall not promote copying from textbook or from others work and shall encourage self/independent and group learning)

A. Measurable

a) Assignments (in writing and doing forms on the aspects of syllabus content and outside the syllabus content. Shall be individual and challenging).
b) Student seminars (on topics of the syllabus and related aspects (individual activity))
c) Quiz (on topics where the content can be compiled by smaller aspects and data (Individuals or groups as teams))
d) Study projects (by very small groups of students on selected local real- time problems pertaining to syllabus or related areas. The individual participation and contribution of students shall be

ensured (team activity))

B. General

a. Group Discussion
b. Others

RECOMMENDED CONTINUOUS ASSESSMENT METHODS:

Some of the following suggested assessment methodologies could be adopted;

1. The oral and written examinations (Scheduled and surprise tests),
2. Closed-book and open-book tests,
3. Programming exercises,
4. Practical assignments and laboratory reports,
5. Observation of practical skills,
6. Individual and group project reports.
7. Efficient delivery using seminar presentations,
8. Viva voce interviews.
9. Computerized adaptive testing, literature surveys and evaluations,
10. Peers and self-assessment, outputs form individual and collaborative work.

@@@@

PARVATHANENI BRAHMAYYA
SIDDHARTHA COLLEGE OF ARTS & SCIENCE
Autonomous
Siddhartha Nagar, Vijayawada–520010
Re-accredited at 'A+' by the NAAC

MODEL PAPER

TITLE: Python Programming **COURSE CODE: 23DAMAL121**

SECTION: B.Sc. (Honours) Data Analytics **SEMESTER: II**

TIME: 3 Hrs. **MAX: 70M**

NOTE TO PAPER SETTER: IN SECTION A & SECTION B, FOR EACH QUESTION ONE SUB QUESTION (A) MUST BE A PROGRAM MEANT FOR LOGICAL TESTING AND ANOTHER SUB QUESTION (B) IS MEANT FOR DESCRIPTIVE / LOGICAL.

SECTION A

ANSWER THE FOLLOWING QUESTIONS. **5 X 4 = 20 Marks**

1. a) Write different types of programming paradigms. (CO1, L1)

   OR

   b) Develop a python code to print sum and average marks of three subjects of a student. (CO1, L1)
2. a) Write a python program to find the factorial of a given number. (CO2, L1)

   OR

   b) Explain if-elif-else statement in python with example (CO2, L2)
3. a) Develop a python program for Lambda function. (CO3, L3)

   OR

   b) Explain about global and local variables in python. (CO3, L2)
4. a) Develop a python program to insert elements into a list, remove elements from a list and sort elements of the list. (CO4, L3)

   OR

   b) Compare lists and tuples in python. (CO4, L3)
5. a) Develop a python program to demonstrate Inheritance. (CO5, L3)

   OR

   b) Explain about classes and objects in python. (CO5, L2)

SECTION B

ANSWER THE FOLLOWING QUESTIONS . **5 X 10 = 50 Marks**

6. a) List and explain data types in python. (CO1, L2)

   OR

   b) Write names of operators in python and explain them with examples. (CO1, L2)
7. a) Develop a python program to find whether the number is palindrome or not.(CO2, L3)

   OR

   b) Explain loops in Python with examples. (CO2, L2)
8. a) Explain different categories of arguments used in functions in python.(CO3, L3)

   OR

   b) Develop a python program to create a module and import it in another program.(CO3, L2)
9. a) Develop a python program to create a dictionary and add, modify, delete values in the dictionary and print them. (CO4, L2)

   OR

   b) Define tuple. Explain about tuples with examples. (CO4, L3)
10. a) Develop a python program to demonstrate exception handling. (CO5, L3)

    OR

    b) Define method overriding. Explain with an example. (CO5, L2)

***

PARVATHANENI BRAHMAYYA
SIDDHARTHA COLLEGE OF ARTS & SCIENCE
Autonomous
Siddhartha Nagar, Vijayawada–520010
Re-accredited at 'A+' by the NAAC

23DAMAP121: Python Programming Lab

Programme: B. Sc. Hons. (Data Analytics) **Semester: II**
Teaching Periods: 30 **No. of Credits:1**

COURSE OUTCOME NO	Upon successful completion of this course, the student will be able to	PROGRAM OUTCOME NO
CO1	Write, Test and Debug Python Programs	**PO1**
CO2	Implement Conditionals and Loops for Python Programs	**PO1,PO2,PO3**
CO3	Organize code into modules for better code organization and reusability.	**PO2,PO3**
CO4	Use functions and represent Compound data using Lists, Tuples and Dictionaries	**PO3,PO4**
CO5	Implement OOP concepts and write applications in python.	**PO5,PO3**

CO-PO MAPPING MATRIX

	CO - PO	PO1	PO2	PO3	PO4	PO5	PO6
23DAMAP121	**CO1**	H					
	CO2	L	H	M			
	CO3		H	M			
	CO4			L	H		
	CO5			H		L	

List of Experiments:

1. Python Program to Find the Square Root
2. Python Program to Swap Two Variables
3. Python Program to Generate a Random Number
4. Python Program to Check if a Number is Odd or Even
5. Python Program to Find the Largest Among Four Numbers
6. Python Program to Check Prime Number
7. Python Program to Check Whether a number is Palindrome or Not
8. Python Program to Display the multiplication Table
9. Python Program to Print the Fibonacci sequence

10. Python Program to Check Armstrong Number
11. Python Program to Find the Sum of Natural Numbers
12. Python Program to Find Factorial of Number Using Recursion
13. Python Program to check given number is prime or not using functions.
14. Python Program to demonstrate usage of keyword, default and variable length arguments.
15. Python Program for lambda functions.
16. Python Program to create module and import it.
17. Python Program to create a list and perform operations on its contents.
18. Python Program to perform operations on tuples.
19. Python Program to create a dictionary and print its content.
20. Python program to perform operations on sets.
21. Python Program for inheritance.
22. Python Program for method overriding.
23. Python Program for exception handling.

Model Paper : Practicals

23DAMAP121: Python Programming Lab

SECTION: B.Sc. (Honours) Data Analytics **SEMESTER: II**

Max. Marks : 50 (CIA: 15 + SEE: 35) **Hrs/Week: 2**

Time: 3 Hrs **Max. Marks : 35**

Section - A

1. Experiment 1	15 M
2. Experiment 2	10 M

Section – B

Viva Voce	10 M
CONTINUOUS ASSESMENT(Internal)	**15 M**
Total	50 M

@@@

PARVATHANENI BRAHMAYYA
SIDDHARTHA COLLEGE OF ARTS & SCIENCE
Autonomous
Siddhartha Nagar, Vijayawada–520010
Re-accredited at 'A+' by the NAAC

23DAMIL121: Python Programming **Year of introduction: 2023-24**
Offered to : ALL UG PROGRAMMES **Course Type : Minor 1 (TH)**
Teaching Periods: 60 **Semester - II** **No. of Credits:3**

Course OBJECTIVE: After taking the course, students will be able to use Python program a Scripting language and Exposure of various problems soling approaches of computer

COURSE OUTCOME NO	Upon successful completion of this course, the student will be able to	PROGRAM OUTCOME NO
CO1	Learn about concepts of programming and python	**PO1, PO2**
CO2	Understand the Decision making and looping controls available in Python Programs	**PO2, PO3**
CO3	Determine the process of using functions and modules	**PO3, PO4**
CO4	Implement the Data structures using Lists, Tuple, Dictionaries	**PO4, PO3**
CO5	Interpret the OOPs concept in Python.	**PO5, PO3**

CO-PO MAPPING MATRIX

	CO - PO	PO1	PO2	PO3	PO4	PO5	PO6
23DAMIL121	**CO1**	H	M				
	CO2		H	M			
	CO3			H	L		
	CO4			L	H		
	CO5			H		L	

UNIT – I: Introduction, Data types, Operators: **12 Periods**

Introduction to Programming: Languages, Generations, Programming Paradigms, Debugging and Testing Approaches.
Python : Introduction, History of Python, Features of Python, Writing and executing python programs, constants, variables, reserved words, input operation, indentation,
Data types in python: integer, string, Boolean, Operators and expressions.

UNIT – II: Decision making and looping **13 Periods**

Control Flow: **Selection or conditional branching** : if, if-else, if-elif- else, nested if.
Loop or iterative statements: for, while, break, continue, pass. Example programs on control flow.

UNIT – III: Functions and Modules: **12 Periods**

Defining Functions, Calling Functions, Passing Arguments, Keyword Arguments, Default Arguments, Variable-length arguments, Lambda or Anonymous Functions, Global and Local Variables.
Modules: Creating modules, import statement, from import statement.

UNIT – IV: Data Structures: **13 Periods**

Lists : Creation, accessing values, updating values, list operations, list methods.
Tuple: tuple creation, accessing values, deleting values, tuple operations. **Sets:** creation, set operations.
Dictionaries: creation, accessing values, adding , modifying , deleting items, built-in dictionary methods.

UNIT – IV: Object Oriented Programming in Python: **10 Periods**

Introduction to OOP, Features of OOP, Merits and Demerits, Classes and Objects, Class method and self-Argument, Public and Private, the init method(constructor), Inheritance, polymorphism and Method Overriding.
Error and Exception handling: Handling Exception using try-except block, Raising Exceptions, User Defined Exceptions.

TEXT BOOKS

Python Programming: Using Problem Solving approach, Reema Thareja, Oxford University Press 2017

REFERENCE BOOKS:

PYTHON PROGRAMMING A Modern Approach, Vamsi Kurama, Pearson Publications, 2017

Recommended Co – Curricular Activities:

(Co-curricular activities shall not promote copying from textbook or from others work and shall encourage self/independent and group learning)

A Measurable

e. Assignments (in writing and doing forms on the aspects of syllabus content and outside the syllabus content. Shall be individual and challenging).
f. Student seminars (on topics of the syllabus and related aspects (individual activity))
g. Quiz (on topics where the content can be compiled by smaller aspects and data (Individuals or groups as teams))
h. Study projects (by very small groups of students on selected local real- time problems pertaining to syllabus or related areas. The individual participation and contribution of students shall be ensured (team activity))

B General

c. Group Discussion

d. Others

RECOMMENDED CONTINUOUS ASSESSMENT METHODS:

Some of the following suggested assessment methodologies could be adopted;

1. The oral and written examinations (Scheduled and surprise tests),
2. Closed-book and open-book tests,
3. Programming exercises,
4. Practical assignments and laboratory reports,
5. Observation of practical skills,
6. Individual and group project reports.
7. Efficient delivery using seminar presentations,
8. Viva voce interviews.
9. Computerized adaptive testing, literature surveys and evaluations,
10. Peers and self-assessment, outputs form individual and collaborative work.

@@@@

PARVATHANENI BRAHMAYYA
SIDDHARTHA COLLEGE OF ARTS & SCIENCE
Autonomous
Siddhartha Nagar, Vijayawada–520010
Re-accredited at 'A+' by the NAAC

MODEL PAPER

23DAMIL121 : PYTHON PROGRAMMING **SEMESTER: II**
TIME: 3 Hrs. **Max.Marks : 70M**

Note to paper setter: In Section A & Section B, for each question one sub question (A) must be a program meant for logical testing and another sub question (B) is meant for descriptive / Logical.

SECTION A

ANSWER THE FOLLOWING QUESTIONS. **5 X 4 = 20 Marks**

1. a) Write different types of programming paradigms. (CO1, L1)

   OR

   b) Develop a python code to print sum and average marks of three subjects of a student. (CO1, L1)
2. a) Write a python program to find the factorial of a given number. (CO2, L1)

   OR

   b) Explain if-elif-else statement in python with example (CO2, L2)
3. a) Develop a python program for Lambda function. (CO3, L3)

   OR

   b) Explain about global and local variables in python. (CO3, L2)
4. a) Develop a python program to insert elements into a list, remove elements from a list and sort elements of the list. (CO4, L3)

   OR

   b) Compare lists and tuples in python. (CO4, L3)
5. a) Develop a python program to demonstrate Inheritance. (CO5, L3)

   OR

   b) Explain about classes and objects in python. (CO5, L2)

SECTION B

ANSWER THE FOLLOWING QUESTIONS . **5 X 10 = 50 Marks**

6. a) List and explain data types in python. (CO1, L2)

   OR

   b) Write names of operators in python and explain them with examples. (CO1, L2)
7. a) Develop a python program to find whether the number is palindrome or not. (CO2, L3)

   OR

   b) Explain loops in Python with examples. (CO2, L2)
8. a) Explain different categories of arguments used in functions in python. (CO3, L3)

   OR

   b) Develop a python program to create a module and import it in another program. (CO3, L2)
9. a) Develop a python program to create a dictionary and add, modify, delete values in the dictionary and print them. (CO4, L2)

   OR

   b) Define tuple. Explain about tuples with examples. (CO4, L3)
10. a) Develop a python program to demonstrate exception handling. (CO5, L3)

    OR

    b) Define method overriding. Explain with an example. (CO5, L2)

@@@@

PARVATHANENI BRAHMAYYA
SIDDHARTHA COLLEGE OF ARTS & SCIENCE
Autonomous
Siddhartha Nagar, Vijayawada–520010
Re-accredited at 'A+' by the NAAC

23DAMIP121: Python Programming Lab **Year of Introduction: 2023-24**
Offered to :ALL UG PROGRAMMES **Course Type : Minor 1(P) Teaching**
Periods: 30 **Semester - II** **No. of Credits:1**

COURSE OUTCOME NO	Upon successful completion of this course, the student will be able to	PROGRAM OUTCOME NO
CO1	Write, Test and Debug Python Programs	**PO1**
CO2	Implement Conditionals and Loops for Python Programs	**PO1,PO2,PO3**
CO3	Organize code into modules for better code organization and reusability.	**PO2,PO3**
CO4	Implement functions and represent Compound data using Lists, Tuples and Dictionaries	**PO3,PO4**
CO5	Implement OOP concepts and write applications in python.	**PO5,PO3**

CO-PO MAPPING MATRIX

	CO - PO	PO1	PO2	PO3	PO4	PO5	PO6
23DAMIP121	**CO1**	H					
	CO2	L	H	M			
	CO3		H	M			
	CO4			L	H		
	CO5			H		L	

List of Experiments:

11. Python Program to Find the Square Root
12. Python Program to Swap Two Variables
13. Python Program to Generate a Random Number
14. Python Program to Check if a Number is Odd or Even
15. Python Program to Find the Largest Among Four Numbers
16. Python Program to Check Prime Number
17. Python Program to Check Whether a number is Palindrome or Not
18. Python Program to Display the multiplication Table

19. Python Program to Print the Fibonacci sequence
20. Python Program to Check Armstrong Number
21. Python Program to Find the Sum of Natural Numbers
22. Python Program to Find Factorial of Number Using Recursion
23. Python Program to check given number is prime or not using functions.
24. Python Program to demonstrate usage of keyword, default and variable length
25. arguments.
26. Python Program for lambda functions.
27. Python Program to create module and import it.
28. Python Program to create a list and perform operations on its contents.
29. Python Program to perform operations on tuples.
30. Python Program to create a dictionary and print its content.
31. Python program to perform operations on sets.
32. Python Program for inheritance.
33. Python Program for method overriding.
34. Python Program for exception handling.
35. Python Program to demonstrate exception handling.
36. Python Program to demonstrate user defined exception.

Question Paper Pattern for Practical Courses

23DAMIP121: Python Programming **SEMESTER: II**
Max. Marks : 50 (CIA: 15 + SEE: 35)

Model Paper : Practicals

Time: 3 Hrs **Max. Marks : 35**

Section - A

1. Experiment 1	15 M
2. Experiment 2	10 M

Section – B

Viva Voce	10 M
CONTINUOUS ASSESMENT(Internal)	**15 M**
Total	50 M

@@@@

PARVATHANENI BRAHMAYYA
SIDDHARTHA COLLEGE OF ARTS & SCIENCE
Autonomous
Siddhartha Nagar, Vijayawada–520010
Re-accredited at 'A+' by the NAAC

23DSMIL121: Python Programming **Year of introduction: 2023-24**
Offered to : ALL UG PROGRAMMES **Course Type : Minor 1(TH)**
Teaching Periods: 60 **Semester - II** **No. of Credits:3**

Course OBJECTIVE: After taking the course, students will be able to use Python program a Scripting language and Exposure of various problems soling approaches of computer

COURSE OUTCOME NO	Upon successful completion of this course, the student will be able to	PROGRAM OUTCOME NO
CO1	Learn about concepts of programming and python	**PO1, PO2**
CO2	Understand the Decision making and looping controls available in Python Programs	**PO2, PO3**
CO3	Determine the process of using functions and modules	**PO3, PO4**
CO4	Implement the Data structures using Lists, Tuple, Dictionaries	**PO4, PO3**
CO5	Interpret the OOPs concept in Python.	**PO5, PO3**

CO-PO MAPPING MATRIX

	CO - PO	PO1	PO2	PO3	PO4	PO5	PO6
23DSMIL121	**CO1**	H	M				
	CO2		H	M			
	CO3			H	L		
	CO4			L	H		
	CO5			H		L	

UNIT – I: Introduction, Data types, Operators: **12 Periods**

Introduction to Programming: Languages, Generations, Programming Paradigms, Debugging and Testing Approaches.
Python : Introduction, History of Python, Features of Python, Writing and executing python programs, constants, variables, reserved words, input operation, indentation,
Data types in python: integer, string, Boolean, Operators and expressions.