PART A: Introduction

Program: **Degree**	Class: **B.Sc.**	Year: **III Year**	Session: **2023-24**

		Subject: **Computer Science**	
1.	Course Code	**S3-COSC2T**	
2.	Course Title	**Data Analysis and Visualization with Python (Theory)**	
3.	Course	**Minor / Elective**	
4.	Pre-Requisite (if any)	To study this course, a student must have successfully completed the course on Programming at Certificate/Diploma Levels.	
5.	Course Learning Outcomes (CLO)	**After studying this subject, students shall be able to –** • Interpret the fundamental Python syntax and semantics and be fluent in the use of Python control flow statements. • Express proficiency in the handling of strings, functions and file handling. • Determine the methods to create and manipulate Python programs by utilizing the data structures like lists, dictionaries, tuples and sets. • Develop proficiency in using NumPy for data manipulation. • Create a variety of data visualizations using Matplotlib. • Apply NumPy and Matplotlib to analyze and visualize real-world datasets. • Complete a hands-on project involving data manipulation and visualization.	
6.	Credit Value	**Theory - 4 Credits**	
7.	Total Marks	Max. Marks: **30+70**	Min. Passing Marks: **35**

PART B: Content of the Course

No. of Lectures (in hours per week): **2 Lectures**

Total No. of Lectures: **60 Hrs.**

Module	Topics	No. of Lectures
I	**Python Basics**: Python interpreter, Python idle, dynamically typed and strongly typed features, basic data types, variables, expressions, statements, operators, flow of execution. Input and Output statements, Conditionals: Boolean values and operators, conditional (if), alternative (else), chained conditional (if-elif-else). Iteration: while, for, break, continue, pass, implementing 'for' through range(), 'in' and 'not in' operators for sequence traversal. Creating and executing .py scripts. **Keywords:** *interpreter, while, for, break, continue, scripts.*	12
II	**Data Structures**: Lists- append, extend, insert, index, remove, pop, count, sort, reverse, slicing, list comprehension, Copying a list: deep copy, shallow copy. Tuples- index, count, usage, use of tuples as a swap function. Dictionaries-keys, values, tuples, nested dictionaries, dictionary comprehension. Strings- Single line	14

Chairman, Central Board of Studies, Computer Science

V	and multi-line strings, formatter, isdigit, isalpha, isalnum, islower, istitle, isspace, title, lower, upper, strip, split, splitlines, join etc. Sets – union, intersection, subset, superset, difference, symmetric difference, copy, add, remove, discard etc. **Functions & File Handling**: Inbuilt Functions- id, len, chr, ord etc., defining and calling a function, arguments, global versus local variables, defining and using lambda functions, the map(), filter(), reduce() functions. **Keywords:** *index, sort, deep copy, tuples, dictionary, sets, strings, function, calling a function, arguments, global variables.*	
III	**NumPy**: Introduction to Numpy, NumPy array in Python, Basics of NumPy Arrays, comparison of Python Lists with Numpy Arrays. **Array**: Array Creation, The Arrange Method, The Zero Method, Numpy array filled with all ones, The linspace Method, The eye Method, Numpy Meshgrid function, empty and full NumPy array, Numpy array filled with all zeros, Numpy array filled with all ones, 2-D Gaussian array, Creating vector in Python using NumPy. Array Indexing, Array Slicing, Data Types, Copy vs View, Array Shape, Array Reshape, Array Iterating, Array Join, Array Split, Array Search, Array Sort, Array Filter, Concatenation of two arrays, Splitting and Comparison of Arrays. Binary Operations, Mathematical Function, String Operations. **Keywords:** *NumPy Arrays, Array Reshape, NumPy Functions, Array Search, Gaussian Array.*	14
IV	**Matrix in NumPy:** Matrix manipulation in Python, empty() function, zeros() function, ones() function, eye() function, identity() function, Adding and Subtracting Matrices in Python. Vector Multiplication, Dot product of two arrays. **Operations on NumPy Array:** Broadcasting with NumPy Arrays, Sorting, Searching and Counting of NumPy arrays. Variations in different Sorting techniques in Python. **Universal Functions:** Creation of ufunc, Simple Arithmetic, Rounding, Trigonometric, Hyperbolic, Set functions. **Keywords:** *NumPy Matrix, Broadcasting with NumPy Array, NumPy ufunc.*	12
V	**Data Visualization with Matplotlib:** Overview of Matplotlib and its capabilities, Creating line plots and scatter plots, Customizing: labels, titles, colors, legends, Creating bar plots and histograms, Adding annotations and text to plots, Creating subplots and multiple plots, Saving and exporting plots. **Advanced Data Visualization**: Creating pie charts and box plots, Visualizing 3D data with Matplotlib, Interactive visualization using widgets. **Keywords:** Matplotlib Charts, *Interactive Visualization, 3-D Plots with Matplolib, subplots, multiplos.*	08
	PART C: Learning Resources	
	Textbooks, Reference Books, Other Resources	
Suggested Readings		

Chairman, Central Board of Studies, Computer Science

Textbooks:
- Taneja Sheetal & Kumar Naveen, "Python Programming: A modular approach", Pearson.
- Liang Y. Daniel, "Introduction to Programming Using Python", Pearson.
- Andreas C. Müller and Sarah Guido," Introduction to Machine Learning with Python: A Guide for Data Scientists,"

Reference Books:
- Zed A. Shaw, "Learn Python the Hard Way", Zed Shaw's Hard Way Series.
- Charles Dierbach, "Introduction to Computer Science using Python", Wiley.
- Michael T. Goodrich, "Data Structures and Algorithms in Python", Wiley.
- Mark Lutz and David Ascher, "Learning Python".
- Phuong Vo.T.H, Martin Czygan, Ashish Kumar, Kirthi Raman, "Python: Data Analytics and Visualization".
- William McKinney, "Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython 2nd Edition""".

Suggestive digital platform web links
https://www.guru99.com/how-to-install-python.html
https://www.python.org/about/gettingstarted/
https://spoken-tutorial.org/media/videos/89/Python-3.4.3-Instruction-Sheet-English.pdf
https://www.learnpython.org/
https://www.w3schools.com/python/

Suggested equivalent online courses
https://nptel.ac.in/courses/106/106/106106145/
https://www.youtube.com/watch?v=rfscVS0vtbw
https://onlinecourses.swayam2.ac.in/aic20_sp33/preview
https://www.w3schools.com/python/numpy/default.asp
https://www.geeksforgeeks.org/python-numpy/

PART D: Assessment and Evaluation

Suggested Evaluation Methods:
Maximum Marks: 100
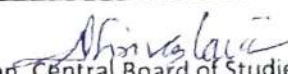**Continuous Comprehensive Evaluation (CCE): 30 Marks**   **University Exam (UE): 70 Marks**

Internal Assessment : Continuous Comprehensive Evaluation (CCE)	Class Tests/ Presentation / Assignment	30 **Marks**
External Assessment: University Exam (UE): Time: **03.00 Hours**	Section (A): Very Short Questions Section (B): Short Questions Section (C): Long Questions	70 **Marks**

Any remarks/suggestions:

Chairman, Central Board of Studies, Computer Science

PRACTICAL

PART A: Introduction

Program: **Degree**	Class: **B.Sc.**	Year: **III Year**	Session: **2023-24**

		Subject: **Computer Science**
1.	Course Code	**S3-COSC2P**
2.	Course Title	**Data Analysis and Visualization with Python (Practical)**
3.	Course Type (Core Course/Elective/Generic Elective/ Vocational	**Minor / Elective**
4.	Pre-Requisite (if any)	To study this course, a student must have successfully completed the course on Programming at Certificate/Diploma Levels.
5.	Course Learning Outcomes (CLO)	**After studying this subject, students shall be able to –** • Understand the python environment and its text editor. • Code and run the programs. • Debug the program. • Interpret the fundamental Python syntax and semantics and be fluent in the use of Python control flow statements. • Determine the methods to create and manipulate Python programs. • Develop proficiency in data manipulation. • Create a variety of data visualizations using Matplotlib.
6.	Credit Value	**Practical - 2 Credits**
7.	Total Marks	Max. Marks: **100** Min. Passing Marks: **35**

PART B: Content of the Course

No. of Lab. Practicals (in hours per week): **1 hour**

Total No. of Lab.: **30Hrs.**

Suggestive List of Practicals	No. of Labs.
1. Find all numbers which are multiple of 17, but not the multiple of 5, between 2000 and 2500. 2. Print the first 2 and last 3 characters in a given string. Use the string slicing. 3. Write a program that eliminates duplicates in a list. 4. Implement shallow copy and deep copy of a list. 5. Find the largest of n numbers, using a user defined function largest() 6. Write a function that capitalizes all vowels in a string. 7. Read a line containing digits and letters. Write a program to give the count of digits and letters. 8. Write a function myReverse() which receives a string as an input and returns the reverse of the string. 9. Use the list comprehension methodology in python, to generate the squares of all odd numbers in a given list. 10. Create a NumPy array with values from 1 to 20. 11. Create a 3x3 identity matrix using NumPy. 12. Generate an array of 10 random integers between 0 and 100. 13. Calculate the mean, median, and standard deviation of an array.	30

Chairman, Central Board of Studies, Computer Science

14. Reshape a 1D array into a 2D array.
15. Filter even numbers from an array using boolean indexing.
16. Calculate the dot product of two matrices.
17. Normalize an array to have values between 0 and 1.
18. Calculate the sum along both rows and columns of a 2D array.
19. Perform element-wise multiplication and division between arrays.
20. Generate a dataset of x and y values and plot it.
21. Analyze and visualize a simple data set (e.g., student grades) using both libraries.
22. Create a bar plot showing comparison of data from two different sources.
23. Visualize data from a CSV file using NumPy and Matplotlib.
24. Generate a contour plot of a 2D function.
25. Analyze and visualize trends in a dataset over time.
26. Create an interactive plot using Matplotlib's interactive mode.

PART C: Learning Resources

Textbooks, Reference Books, Other Resources

Suggested Readings

Textbooks:

- Taneja Sheetal & Kumar Naveen, "Python Programming: A modular approach", Pearson.
- Liang Y. Daniel, "Introduction to Programming Using Python", Pearson.

Reference Books:

- Zed A. Shaw, "Learn Python the Hard Way", Zed Shaw's Hard Way Series
- Charles Dierbach, "Introduction to Computer Science using Python", Wiley
- Michael T. Goodrich, "Data Structures and Algorithms in Python", Wiley
- Mark Lutz and David Ascher, "Learning Python".
- Phuong Vo.T.H, Martin Czygan, Ashish Kumar, Kirthi Raman, "Python: Data Analytics and Visualization".
- William McKinney, "Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython 2nd Edition""".

Suggestive digital platform web links

https://www.guru99.com/how-to-install-python.html
https://www.python.org/about/gettingstarted/
https://spoken-tutorial.org/media/videos/89/Python-3.4.3-Instruction-Sheet-English.pdf
https://www.learnpython.org/
https://www.w3schools.com/python/

Suggested equivalent online courses

https://nptel.ac.in/courses/106/106/106106145/
https://www.youtube.com/watch?v=rfscVS0vtbw
https://onlinecourses.swayam2.ac.in/aic20_sp33/preview
https://www.w3schools.com/python/numpy/default.asp
https://www.geeksforgeeks.org/python-numpy/

PART D: Assessment and Evaluation

Internal Assessment:		External Assessment:	
Class Interaction/Quiz		Viva voce practical	
Attendance	30	Practical record file	70
Assignments (Charts/ Model)/ Technology Dissemination/ Excursion/ Lab visit/ Industrial Training		Table work / Experiments	

Total Marks: 100

Any remarks/ suggestions:

Chairman, Central Board of Studies, Computer Science