



NATIONAL INSTITUTE OF TECHNOLOGY, TIRUCHIRAPPALLI
DEPARTMENT OF COMPUTER APPLICATIONS

COURSE PLAN – PART I			
Name of the programme and specialization	Master of Computer Applications		
Course Title	Problem solving and Programming		
Course Code	CA11	No. of Credits	3
Course Code of Pre-requisite subject(s)	-		
Session	July 2022	Section	A & B
Name of Faculty	Dr. S. Sangeetha	Department	Computer Applications
Official Email	sangeetha@nitt.edu	Telephone No.	0431-2503743
Name of PAC Chairperson	Dr.U.Srinivasulu Reddy		
Official E-mail	usreddy@nitt.edu	Telephone No.	---
Course Type (please tick appropriately)	<input checked="" type="checkbox"/> Core course <input type="checkbox"/> Elective course		
Syllabus (approved in BoS)			
<p>Programming paradigms- Program Development Cycle- Evolution of Programming languages - Computational Problem Solving - Principles of Structured programming: Sequential, selective and repetitive structures- Modular Programming: Functions and Procedures-Algorithms.</p> <p>Introduction to Python Programming: Python interpreter- Garbage collection- Python working Environment-Mutable and Immutable objects- Variables- Dynamic typing- expressions- Operators: precedence and Associativity- comments- Conditionals: conditional - alternative - chained conditional - Short Circuits; Iteration: while- for-range- break- continue- pass; Strings: string slices- immutability- string methods- Regular Expression: Patterns- Matching- Search and replace.</p> <p>Lists: Traversing a List- list operations- list slices- list methods- list loop- mutability- aliasing- cloning lists- list parameters; Tuples: tuple assignment- tuple as return value. Dictionary: operations and methods- Tuples as key; Set: Creation- Methods. Comprehension: List comprehension and Dictionary comprehensions; Map- Filter and Reduce.</p> <p>Functions: Definitions- parameters and arguments: Keyword arguments- Positional arguments- Parameter unpacking- Scope: Local, Global and Enclosed. Recursion- Lambda functions- Higher order Functions; Object orientation in Python: Classes- Objects.</p> <p>Files and exception: text and binary files- CSV files- JSON Files- reading and writing files- Object serialization; Exception Handling: Errors and exceptions- handling exceptions- modules- packages: Creating modules and packages- Python standard Library: OS- Sys- Collections- Random- Library for Data science: PANDAS- NUMPY.</p>			

REFERENCES:

1. John V. Guttag, Introduction to Computation and Programming Using Python: with Application to Computational Modelling and Understanding Data- Third Edition- MIT press- 2021.
2. Paul J. Deitel- Harvey Deitel- Python for Programmers- First Edition- Pearson- 2020.
3. Martin C. Brown- Python: The Complete Reference-Fourth Edition-Mc-Graw Hill- 2018.
4. Robert Sedgewick- Kevin Wayne- Robert Dondero- —Introduction to Programming in Python: An Inter-disciplinary Approach- First Edition-Pearson India- 2016.
5. Allen B. Downey- Think Python: How to Think like a Computer Scientist- 2nd edition- Updated for Python 3- O'Reilly- 2015.

COURSE OBJECTIVE

To learn problem solving methodologies and aspects of Python programming.

MAPPING OF COs with POs

Course Outcomes Students will be able to	Programme Outcomes (PO) (Enter Numbers only)
1. Write structured pseudo codes for a given problem.	1,2,3,5
2. Develop Python programs with conditionals- loops and data structures	4,5
3. Design and create Python applications using functions and files	1,2,3,4,5


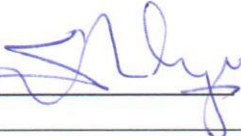

COURSE PLAN – PART II**COURSE OVERVIEW**

This course introduces the concepts of computational problem solving to the students using python. It begins with the basics of writing algorithms for the problems. It introduces the syntax of Python and the various data structures such as lists, dictionary and tuples along with the methods and comprehension. The course also provides clear description string manipulation, using regular expressions to identify patterns as well as handling of files. It comprises the concepts functions and Higher order functions to solve real world problems. It also introduces problem solving using object orientation in python. The course also presents third party libraries such Numpy, Pandas that support recent computing technologies like data analytics.

COURSE TEACHING AND LEARNING ACTIVITIES

Week	Contact Hours	Topic	Mode of Delivery
1	1	Introduction to Problem solving Program Development Cycle	Presentation
	2	Evolution of Programming languages Computational Problem Solving	Presentation
	3	Sequential, selective structures	Presentation
	4	Repetitive control structures	Chalk & talk
2	1	Algorithms	Presentation
	2	Modular Programming: Functions and Procedures	Chalk & talk
	3	Writing Algorithms for a given problem	Chalk & talk
	4	Writing Algorithms for a given problem	Chalk & talk
	1	Writing Algorithms for a given problem	Chalk & talk
3	2	Introduction: Python interpreter, Garbage collection, Python working Environment	Presentation

	3	Mutable and Immutable objects, Variables, Dynamic typing, expressions	Presentation
	4	Operators: Precedence and Associativity	Chalk & talk
4	1	Conditional, Chained conditional, Short Circuits	Presentation
	2	Iteration: while, for, range, break, continue, pass	Problem solving
	3	Strings: string slices, immutability, string methods	Demo & Presentation
	4	Problem solving	Demo & Problem Solving
5	1	Regular Expression Patterns, Matching, Search and replace	Presentation
	2	Regular Expression (Re) Module	Demo
	3	Regular Expression Patterns	Problem Solving
	4	Problem Solving	Problem Solving
6	1	Lists: Traversing a List, List slices, list methods, list loop, mutability, aliasing, cloning lists	Presentation & Demo
	2	Tuples: tuple assignment, tuple as return value. Set: Creation, Methods	Presentation & Demo
	3	Dictionary: operations and methods	Presentation & Demo
	4	Problem Solving	Problem Solving
7	1	List comprehension	Demo
	2	List comprehension	Demo
	3	Dictionary comprehensions, Map, Filter and Reduce.	Demo
	4	Problem Solving	Problem Solving
8	1	Functions	Demo
	2	Keyword arguments, Positional argument, Parameter unpacking	Demo
	3	Scope: Local, Global and Enclosed scope	Demo
	4	Problem Solving	Problem Solving
9	1	Recursion	Demo
	2	Lambda functions	Presentation
	3	Higher order Functions	Presentation
	4	Problem Solving	Problem Solving
10	1	Object orientation concepts	Presentation
	2	Classes and Objects	Presentation & Demo
	3	Classes and Objects	Presentation & Demo
	4	Problem Solving	Problem Solving
11	1	Files: text and binary files	Presentation & Demo
	2	CSV files- JSON Files	Demo
	3	Errors and exceptions, handling exceptions	Presentation
	4	Python standard Library: OS, Collections, Random	Demo
12	1	Python standard Library: Sys, Creating Modules and packages	Presentation & Demo
	2	NUMPY	Demo
	3	PANDAS	Demo
	4	Problem Solving	Problem Solving

COURSE ASSESSMENT METHODS (shall range from 4 to 6)				
S. No.	Mode of Assessment	Week/Date	Duration	% Weightage
1	Test 1	As per Academic Schedule	60 Minutes	15
2	Test 2	As per Academic Schedule	60 Minutes	15
3	Programming Assignment / Project	Week 9	4 Weeks	20
CPA	Compensation Assessment	At the end of the course	60 Minutes	15
4	Final Assessment	At the end of the course	120 Minutes	50
COURSE EXIT SURVEY				
<ul style="list-style-type: none"> The students may give their feedback at any time to the course faculty which will be duly addressed. Email:sangeetha@nitt.edu The students may give their feedback during Class Committee meeting. 				
COURSE POLICY (including compensation assessment to be specified)				
ATTENDANCE POLICY (A uniform attendance policy)				
➤ 75% attendance has to be maintained.				
ACADEMIC DISHONESTY & PLAGIARISM				
<ul style="list-style-type: none"> ➤ Zero mark to be awarded for the offenders. For copying from another student, both students get the same penalty of zero mark. ➤ The departmental disciplinary committee including the course faculty member, PAC chairperson and the HoD, as members shall verify the facts of the malpractice and award the punishment if the student is found guilty. The report shall be submitted to the Academic office. ➤ The students are expected to come out with their original solution for problems given as assignment, and tests/examinations. 				
ADDITIONAL INFORMATION, IF ANY				
➤ The students can meet course faculty on every Tuesday 4 to 5 Pm for clearing their doubts				
FOR APPROVAL				
Course Faculty <u></u> CC- Chairperson <u></u> HOD <u></u>				