NATIONAL INSTITUTE OF TECHNOLOGY, TIRUCHIRAPPALLI

DEPARTMENT OF COMPUTER APPLICATIONS

COURSE PLAN – PART I			
Name of the programme and specialization	B. Tech. / MME		
Course Title	Introduction to Computer Programming (Theory & Lab)		
Course Code	CSIR12	**No. of Credits**	3
Course Code of Pre-requisite subject(s)	NIL		
Session	~~July~~ / January 2021	**Section (if, applicable)**	NA
Name of Faculty	Dr. Jitendra Kumar	**Department**	Computer Applications
Email	jitendra@nitt.edu	**Telephone No.**	0431-2503734
Name of PAC Chairman	Dr. Karthick V		
E-mail	karthikv@nitt.edu	**Telephone No.**	-
Course Type	GIR		

Syllabus (approved in BoS)

Introduction to computers – Computer Organization – Characteristics – Hardware and Software– Modes of operation – Types of programming languages – Developing a program. Algorithms – Characteristics – Flowcharts.

Data types; variables, assignments; immutable variables; numerical types; arithmetic operators and expressions; comments; understanding error messages; Conditions, Boolean logic, logical operators; ranges; Control statements: if-else, loops (for, while); short-circuit (lazy) evaluation

Strings and text files; manipulating files and directories, OS and SYS modules; text files: reading/writing text and numbers from/to a file; creating and reading a formatted file (csv or tab- separated). String manipulations: subscript operator, indexing, slicing a string; strings and number system: converting strings to numbers and vice versa. Binary, octal, hexadecimal numbers

Lists, tuples, and dictionaries; basic list operators, replacing, inserting, removing an element; searching and sorting lists; dictionary literals, adding and removing keys, accessing and replacing values; traversing dictionaries.

Design with functions: hiding redundancy, complexity; arguments and return values; formal vs actual arguments, named arguments- Program structure and design- Recursive functions – Introduction to classes and OOP.

Reference Books

1. Kenneth A. Lambert, *Fundamentals of Python: First Programs*, CENGAGE Learning, 2012.
2. Guido van Rossum and Fred L. Drake Jr, *An Introduction to Python – Revised and updated for Python 3.2*, Network Theory Ltd., 2011.
3. Reema Thareja, *Python Programming using Problem Solving Approach*, Oxford University Press, 2017
4. Allen B. Downey, *Think Python: How to Think Like a Computer Scientist*, 2nd edition, Updated for Python 3, Shroff/O'Reilly Publishers, 2016.
5. John V Guttag, *Introduction to Computation and Programming Using Python*, Revised and expanded Edition, MIT Press, 2013.

COURSE OBJECTIVE(S)

- To learn the fundamentals of computers
- To learn the problem solving techniques using algorithms and procedures
- To read, write and execute simple Python Programs
- To learn and use Python data structures – lists, tuples and dictionaries

COURSE OUTCOMES (CO)

Course Outcomes	**Aligned Programme Outcomes (PO)**
Students will be able to:	
1. Write algorithms for problems	**PO I, II, III**
2. Use syntax and semantics of Python programming language for problem solving	**PO I, II, III, V**
3. Code a given logic in Python language	**PO IV, V**
4. Appreciate and apply appropriate Data structures available in Python language for solving problem	**PO III, IV, V**

COURSE PLAN – PART II

COURSE OVERVIEW

This lab integrated course covers basics of Python programming. It provides insights on problem solving aspects by discussing several examples using algorithms and flowcharts. Students are introduced to major language elements including fundamental data types, flow control, and standard function libraries for developing real time application. Comprehensive hands-on exercises are integrated throughout to reinforce learning and develop real competency.

COURSE TEACHING AND LEARNING ACTIVITIES			
S. No.	**Week/ Contact Hours**	**Topic**	**Mode of Delivery**
1	Week 1 (2 Lectures)	Introduction to computers – Computer Organization – Characteristics - Hardware and Software– Modes of operation – Types of programming languages	Pen and Paper (Online), Power Point Presentation
2	Week 2 (2 Lectures)	Developing a program – Algorithms – Characteristics – Flowcharts	Pen and Paper (Online), Power Point Presentation
3	Week 3 (2 Lectures)	Data types; variables, assignments; immutable variables; numerical types, arithmetic operators and expressions;	Pen and Paper (Online), Power Point Presentation
4	Week 4 (2 Lectures)	Boolean logic, logical operators, comments; understanding error messages; Conditions, ranges;	Pen and Paper (Online), Power Point Presentation
5	Week 5 (2 Lectures)	Control statements: if-else, loops (for, while); short-circuit (lazy) evaluation	Pen and Paper (Online), Power Point Presentation
6	Week 6 (2 Lectures)	Strings and text files; manipulating files and directories, OS and SYS modules	Pen and Paper (Online), Power Point Presentation
7	Week 7 (2 Lectures)	text files: reading/writing text and numbers from/to a file; creating and reading a formatted file (csv or tab- separated)	Pen and Paper (Online), Power Point Presentation
8	Week 8 (2 Lectures)	String manipulations: subscript operator, indexing, slicing a string; strings and number system: converting strings to numbers and vice versa. Binary, octal, hexadecimal numbers	Pen and Paper (Online), Power Point Presentation
9	Week 9 (2 Lectures)	Lists, tuples, and dictionaries; basic list operators, replacing, inserting, removing an element; searching and sorting lists;	Pen and Paper (Online), Power Point Presentation
10	Week 10 (2 Lectures)	dictionary literals, adding and removing keys, accessing and replacing values; traversing dictionaries.	Pen and Paper (Online), Power Point Presentation
11	Week 11 (2 Lectures)	Design with functions: hiding redundancy, complexity; arguments and return values; formal vs actual arguments, named arguments	Pen and Paper (Online), Power Point Presentation
12	Week 12 (2 Lectures)	Program structure and design- Recursive functions – Introduction to classes and OOP.	Pen and Paper (Online), Power Point Presentation

Lab Activities

On each lab, students must implement 3 to 5 questions/programs based on the topics mentioned below:

1	Week 1	Programs using sequential constructs	Virtual Lab
2	Week 2	Programs using selection constructs	Virtual Lab
3	Week 3	Programs using iterative constructs	Virtual Lab
4	Week 4	Programs using nested for loops	Virtual Lab
5	Week 5	Programs using lists	Virtual Lab
6	Week 6	Programs using tuples and dictionaries	Virtual Lab
7	Week 7	Simple Python functions	Virtual Lab
8	Week 8	File input and output	Virtual Lab
9	Week 9	Sorting and searching programs	Virtual Lab
10	Week 10	Recursion	Virtual Lab

COURSE ASSESSMENT METHODS

The assessment in this course has two components, viz., Theory and Practical. The assessment in Theory component has cycle test and final assessment whose details are given in the below table. The assessment in Theory will be done for a total of 70 marks. The assessment in the Practical component has periodical record / observation evaluation and final assessment whose details are given in the below table. The assessment in Practical will be done for a total of 30 marks. The total marks for this course is 100.

Course Assessment Methods: Theory

S. No.	Mode of Assessment	Week/Date	Duration	% Weightage
1	Cycle Test 1	5th Week	60 Minutes	20
2	Cycle Test 2	9th Week	60 Minutes	20

CPA	Compensation Assessment*	As per academic schedule	60 Minutes	20
3	Final Assessment	As per academic schedule	120 Minutes	30
Total Theory Marks				70%
Course Assessment Methods: Practical				
4	Continuous Assessment (Weekly Lab)	--	--	10
5	Cycle Test	5th Week	120 Minutes	10
CPA	Compensation Assessment Lab*	As per academic schedule	120 Minutes	10
6	Final Assessment Practical	As per academic schedule	120 Minutes	10
Total Practical Marks				30%
Total Marks [Theory (70%) + Practical (30%)]				100%

***mandatory; refer to guidelines on page 7**

COURSE EXIT SURVEY (mention the ways in which the feedback about the course shall be assessed)

The students through the class representattive may give their feedback at any time to the course coordinator which will be duly adddressed.
The students may give their feedback during class committee meetings.

COURSE POLICY (including compensation assessment to be specified)

MODE OF CORRESPONDENCE
By Email: jitendra@nitt.edu

COMPENSATION ASSESSMENT POLICY
Compensation assessment will be conducted for absentees in cycle test I or cycle test II only after the submission of medical or On-Duty certificates signed by competent authority.

ATTENDANCE POLICY (A uniform attendance policy as specified below shall be followed)

Institute guidelines will be followed.

ACADEMIC DISHONESTY & PLAGIARISM

- Carrying bits of paper, talking to other students, copying from any other sources during an assessment will be treated as punishable dishonesty.
- 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 above policy against academic dishonesty shall be applicable for all the programmes.
ADDITIONAL INFORMATION
NIL
FOR APPROVAL
Course Faculty _____ **CC-Chairperson** _____ **HOD** _____

Guidelines

a) The number of assessments for any theory course shall range from 4 to 6.

b) Every theory course shall have a final assessment on the entire syllabus with at least 30% weightage.

c) One compensation assessment for absentees in assessments (other than final assessment) is mandatory. Only genuine cases of absence shall be considered.

d) The passing minimum shall be as per the regulations.

B.Tech. Admitted in				P.G.
2018	2017	2016	2015	
35% or (Class average/2) whichever is greater.		(Peak/3) or (Class Average/2) whichever is lower		40%

e) Attendance policy and the policy on academic dishonesty & plagiarism by students are uniform for all the courses.

f) Absolute grading policy shall be incorporated if the number of students per course is less than 10.

g) Necessary care shall be taken to ensure that the course plan is reasonable and is objective.