INSTITUTE OF AERONAUTICAL ENGINEERING

(Autonomous)
Dundigal, Hyderabad -500 043

AERONAUTICAL ENGINEERING

COURSE DESCRIPTOR

Course Title	**Object Oriented Programming through Python Laboratory**				
Course Code	AITB08				
Programme	B.Tech				
Semester	III	AE			
Course Type	Core				
Regulation	IARE - R18				
Course Structure	**Theory**			**Practical**	
	Lectures	**Tutorials**	**Credits**	**Laboratory**	**Credits**
	-	-	-	3	2
Chief Coordinator	Ms. A Jayanthi, Assistant Professor, CSE				
Course Faculty	Mr. N Poornachandra Rao, Assistant Professor, CSE				

I. COURSE OVERVIEW:

The course covers the basics of programming and demonstrates fundamental programming techniques, customs and terms including the most common library functions and the usage of the preprocessor. This course helps the students in gaining the knowledge to write simple Python language applications, mathematical and engineering problems. This course helps to undertake future courses that assume this programming language as a background in python programming. Topics include variables, data types, functions, control structures, lists, strings, class, methods, inheritance, polymorphism, overriding principles. This course in reached to student by power point presentations, lecture notes, and lab involve the problem solving in mathematical and engineering areas.

II. COURSE PRE-REQUISITES:

Level	Course Code	Semester	Prerequisites	Credits
-	-	-	Basic Programming Concepts	-

III. MARKS DISTRIBUTION:

Subject	SEE Examination	CIA Examination	Total Marks
Object Oriented Programming through Python Laboratory	70 Marks	30 Marks	100

IV. DELIVERY / INSTRUCTIONAL METHODOLOGIES:

✘	Chalk & Talk	✘	Quiz	✘	Assignments	✘	MOOCs
✔	LCD / PPT	✘	Seminars	✘	Mini Project	✔	Videos
✔	Open Ended Experiments						

V. EVALUATION METHODOLOGY:

Each laboratory will be evaluated for a total of 100 marks consisting of 30 marks for internal assessment and 70 marks for semester end lab examination. Out of 30 marks of internal assessment, continuous lab assessment will be done for 20 marks for the day to day performance and 10 marks for the final internal lab assessment.

Semester End Examination (SEE): The semester end lab examination for 70 marks shall be conducted by two examiners, one of them being Internal Examiner and the other being External Examiner, both nominated by the Principal from the panel of experts recommended by Chairman, BOS.

The emphasis on the experiments is broadly based on the following criteria:

20 %	To test the preparedness for the experiment.
20 %	To test the performance in the laboratory.
20 %	To test the calculations and graphs related to the concern experiment.
20 %	To test the results and the error analysis of the experiment.
20 %	To test the subject knowledge through viva – voce.

Continuous Internal Assessment (CIA):
CIA is conducted for a total of 30 marks (Table 1), with 20 marks for continuous lab assessment during day to day performance, 10 marks for final internal lab assessment.

Table 1: Assessment pattern for CIA

Component	**Laboratory**		**Total Marks**
Type of Assessment	Day to day performance	Final internal lab assessment	
CIA Marks	20	10	30

Continuous Internal Examination (CIE):
One CIE exams shall be conducted at the end of the 16th week of the semester. The CIE exam is conducted for 10 marks of 3 hours duration.

Preparation	**Performance**	**Calculations and Graph**	**Results and Error Analysis**	**Viva**	**Total**
2	2	2	2	2	10

VI. HOW PROGRAM OUTCOMES ARE ASSESSED:

Program Outcomes (POs)		Strength	Proficiency assessed by
PO 1	**Engineering knowledge**: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.	3	Videos
PO 2	**Problem analysis**: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.	2	Lab Exercises
PO 3	**Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.	3	Case Studies
PO 5	**Modern tool usage**: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.	3	Videos
PO 12	**Life-long learning**: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.	2	Case Studies

3 = High; 2 = Medium; 1 = Low

VII. HOW PROGRAM SPECIFIC OUTCOMES ARE ASSESSED:

Program Specific Outcomes (PSOs)		Strength	Proficiency assessed by
PSO 1	**Professional Skills:** Able to utilize the knowledge of aeronautical/aerospace engineering in innovative, dynamic and challenging environment for design and development of new products.	3	Videos
PSO 2	**Problem-solving Skills:** Imparted through simulation language skills and general purpose CAE packages to solve practical, design and analysis problems of components to complete the challenge of airworthiness for flight vehicles	1	Case Studies
PSO 3	**Practical implementation and testing skills:** Providing different types of in house and training and industry practice to fabricate and test and develop the products with more innovative technologies.	3	Lab Exercises
PSO 4	**Successful Career and Entrepreneurship:** To prepare the students with broad aerospace knowledge to design and develop systems and subsystems of aerospace and allied systems and become technocrats.	3	Case Studies

3 = High; 2 = Medium; 1 = Low

VIII. COURSE OBJECTIVES :

The course should enable the students to:	
I	Learn adequate knowledge to write, test, and debug simple Python programs.
II	Understand programming skills using the fundamentals and basics of Python Language.
III	Improve problem solving skills using strings, and functions.

IV	Understand the compound data using Python lists, class, methods.
V	Understand the concepts of inheritance, polymorphism and overriding.

IX. COURSE OUTCOMES (COs):

COs	Course Outcome	CLOs	Course Learning Outcome
CO 1	Understand the basic concepts of python	CLO 1	Analyze a given problem and develop an algorithm to solve the problem.
		CLO 2	Describe the fundamental programming constructs and articulate how they are used to develop a program.
		CLO 3	Gain knowledge to identify appropriate Python language constructs to write basic programs.
		CLO 4	Identify the right data representation formats based on the requirements of the problem.
CO 2	Understand branching statements, loop statements.	CLO 5	Understand branching statements, loop statements and use them in problem solving.
CO 3	Explore the concepts of functions, lists and multidimensional lists.	CLO 6	Understand the concept of functions.
		CLO 7	Learn data types and use them to solve statistical problems.
		CLO 8	Identify the right string function to write string programs.
		CLO 9	Distinguish Create, run and manipulate Python Programs using core data structures like Lists, multidimensional lists.
CO 4	Understand the concepts of class, object and constructors.	CLO 10	Explain the concept of class and objects.
		CLO 11	Differentiate call by Object and call by Object reference parameter passing mechanisms and constructor.
CO 5	Understand the concepts of inheritance, polymorphism, overriding and event driven programming.	CLO 12	Demonstrate the implementation of inheritance.
		CLO 13	Use polymorphism to process objects depending on their class.
		CLO 14	Understand overriding magic methods.
		CLO 15	Explain the concept of event-driven programming.

X. COURSE LEARNING OUTCOMES (CLOs):

CLO Code	CLO's	At the end of the course, the student will have the ability to:	PO's Mapped	Strength of Mapping
AITB08.01	CLO 1	Analyze a given problem and develop an algorithm to solve the problem	PO 1, PO 2	2
AITB08.02	CLO 2	Describe the fundamental programming constructs and articulate how they are used to develop a program.	PO 1, PO 2	2
AITB08.03	CLO 3	Gain knowledge to identify appropriate Python language constructs to write basic programs.	PO 2, PO 3	3
AITB08.04	CLO 4	Identify the right data representation formats based on the requirements of the problem.	PO 2	2
AITB08.05	CLO 5	Understand branching statements, loop statements and use them in problem solving.	PO 1,PO 2, PO 3	3
AITB08.06	CLO 6	Understand the concept of functions.	PO 1, PO 2 PO 3	3

CLO Code	CLO's	At the end of the course, the student will have the ability to:	PO's Mapped	Strength of Mapping
AITB08.07	CLO 7	Learn data types and use them to solve statistical problems.	PO 1, PO 2, PO 3	3
AITB08.08	CLO 8	Identify the right string function to write string programs.	PO 1, PO 2	2
AITB08.09	CLO 9	Distinguish Create, run and manipulate Python Programs using core data structures like Lists, multidimensional lists.	PO 1, PO 2	2
AITB08.10	CLO 10	Explain the concept of class and objects.	PO 2, PO 3	2
AITB08.11	CLO 11	Differentiate call by Object and call by Object reference parameter passing mechanisms and constructor.	PO 2, PO 3	2
AITB08.12	CLO 12	Demonstrate the implementation of inheritance.	PO 1, PO 2,PO5	3
AITB08.13	CLO 13	Use polymorphism to process objects depending on their class.	PO 1, PO 2,PO3	3
AITB08.14	CLO 14	Understand overriding magic methods.	PO 1, PO 2	2
AITB08.15	CLO 15	Explain the concept of event-driven programming.	PO 1, PO 2	3

3 = High; 2 = Medium; 1 = Low

XI. MAPPING COURSE LEARNING OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Learning Outcomes (CLOs)	Program Outcomes (POs)												Program Specific Outcomes (PSOs)			
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3	PSO4
CLO 1	3	2											2	2		
CLO 2	3	2											3	2		1
CLO 3	1	3	2										1	3		
CLO 4		3										1	3			
CLO 5	2	3	2											3		1
CLO 6		3	2											3		
CLO 7	3	2	1										2	3		1
CLO 8	2	3	1										2	3		
CLO 9	2	3	1		1								1	3	1	
CLO 10		2	3										2	3	1	
CLO 11		2	3										3	2		
CLO 12	3	2			2									3		
CLO 13	3	2	2										2	3		1
CLO 14	2	3											3			
CLO 15	3	2			2								1	1		

3 = High; 2 = Medium; 1 = Low

XII. ASSESSMENT METHODOLOGIES – DIRECT

CIE Exams	PO 1, PO 2, PO 3, PO 5, PO 12,PSO 1,PSO 2,PSO 3,PSO 4	SEE Exams	PO 1, PO 2, PO 3,PO 5, PO 12,PSO 1,PSO 2,PSO 3,PSO 4	Assignments	-	Seminars	-
Laboratory Practices	PO 1, PO 2, PO 3, PO 5, PO 12,PSO 1,PSO 2,PSO 3,PSO 4	Student Viva	PO 2	Mini Project	-	Certification	-

XIII. ASSESSMENT METHODOLOGIES - INDIRECT

✔	Early Semester Feedback	✔	End Semester OBE Feedback
✘	Assessment of Mini Projects by Experts		

XIV. SYLLABUS

LIST OF EXPERIMENTS	
Week-1	**BASICS OF PYTHON**
Write Python programs for the following: a. Purposefully raise Indentation Error and correct it. b. Compute distance between two points taking input from the user (Pythagorean Theorem). c. To takes numbers as command line arguments and print its sum.	
Week-2	**CONTROL FLOW**
Write Python programs for implementing the following: a. Checking whether the given number is even number or not. b. Finding the factorial of a number. c. Print the prime numbers below 100.	
Week-3	**STRINGS**
Write Python programs for implementing the following: a. Count the numbers of characters in the string and store them in a dictionary data structure b. Using split and joins methods in the string and trace a birthday with a dictionary data structure.	
Week-4	**LIST**
Write Python programs to for the following: a. Finding mean, median, mode for the given set of numbers in a list. b. Function dups to find all duplicates in the list.	
Week-5	**MULTI DIMENSIONAL LIST**
Write Python programs for the following: a. Addition of two square matrices. b. Multiplication of two matrices.	
Week-6	**CLASS**
Write Python programs to implement the following: a. Find the validity of a string of parentheses, '(', ')', '{', '}', '[' and ']. These brackets must be close in the correct order, for example "()" and "()[]{}" are valid but "[)", "({[)]" and "{{{" are invalid. b. Get all possible unique subsets from a set of distinct integers.	

Week-7	**METHODS**
Write Python programs to do the following a. Create a Python class named Circle constructed by a radius and two methods which will compute the area and the perimeter of a circle. b. Create a Python class named Rectangle constructed by a length and width and a method which will compute the area of a rectangle.	
Week-8	**CONSTRUCTORS**
Write Python program to implement constructors.	
Week-9	**INHERITANCE**
Write Python program to implement inheritance.	
Week-10	**POLYMORPHISM**
Write Python program to implement Polymorphism.	
WeeK-11	**OVERRIDING MAGIC METHODS**
Write Python program to override Magic Methods.	
WeeK-12	**EVENT-DRIVEN PROGRAMMING**
Write Python program to create a simple calculator, where the user will enter a number in a text field, and either add it to or subtract it from a running total, which we will display. We will also allow the user to reset the total.	
Text Books:	
1. R Nageswara Rao, "Core Python Programming", Dreamtech press, 2017 Edition. 2. Dusty Philips, "Python 3 Object Oriented Programming", PACKT Publishing, 2nd Edition, 2015.	
Reference Books:	
1. Rance D. Necaise, "Object-Oriented Programming in Python Documentation Release 1", University of Cape Town and individual contributors, 2017.	

XV. COURSE PLAN:

The course plan is meant as a guideline. Probably there may be changes.

Week No.	Topics to be covered	Course Learning Outcomes (CLOs)	Reference
1	Basics Of Python	CLO 1, CLO 2, CLO 3, CLO 4	T2:1.1-1.2
2	Control Flow	CLO 5, CLO 6, CLO 7	T2:2.1-2.2
3	Strings	CLO 8, CLO 9	T2:1.4-1.5
4	List	CLO 9	T2:3.1-3.5
5	Multi Dimensional List	CLO 9, CLO 10	T2: 5.2-5.3
6	Class	CLO 10, CLO 11	T2: 6.1-6.6
7	Methods	CLO 11, CLO 12, CLO 13	T2: 6.7
9	Inheritance	CLO 12, CLO 11	T2: 8.1-8.3
10	Polymorphism	CLO 13 , CLO 11	T2: 11.1-11.5

Week No.	Topics to be covered	Course Learning Outcomes (CLOs)	Reference
11	Overriding Magic Methods	CLO 14, CLO 13	T2: 8.1-8.3,8.7-8.8
12	Event-Driven Programming	CLO 15, CLO 13	T2: 11.1-11.5

XVI. GAPS IN THE SYLLABUS - TO MEET INDUSTRY / PROFESSION REQUIREMENTS:

S No	Description	Proposed actions	Relevance with POs	Relevance with PSOs
1	Assist student to design system calls in operating systems.	Seminars	PO 1	PSO 1,PSO2
2	Stimulate students to develop graphics programming.	Seminars / NPTEL	PO 2	PSO 1
3	Encourage students to solve real time applications and prepare towards competitive examinations.	1. Build IT 2. Proficiency Test 3. Coding Hackthon/ Competitions	PO 2	PSO 1,PSO2

Prepared by:
Ms. A Jayanthi, Assistant Professor **HOD, AE**