Course	ENR106 Introduction to Programming	Semester	Monsoon Semester 2024
Faculty Name(s)	Jayendra Bhalodiya, Kuntalkumar Patel	Contact	jayendra.bhalodiya@ahduni.edu.in, kuntal.patel@ahduni.edu.in
School	SEAS	Credits	3
GER Category:	Not Applicable	Teaching Pedagogy Enable:NO	P/NP Course: Can not be taken as P/NP

<table>
<tr><td>Schedule</td><td>
<table>
<tr><td>Section 1</td><td>02:30 pm to 04:00 pm</td><td>Fri</td><td>29-07-24 to 26-11-24</td></tr>
<tr><td></td><td>04:00 pm to 05:30 pm</td><td>Fri</td><td>29-07-24 to 26-11-24</td></tr>
<tr><td></td><td>05:30 pm to 07:00 pm</td><td>Fri</td><td>29-07-24 to 26-11-24</td></tr>
<tr><td>Section 2</td><td>02:30 pm to 04:00 pm</td><td>Fri</td><td>29-07-24 to 26-11-24</td></tr>
<tr><td></td><td>04:00 pm to 05:30 pm</td><td>Fri</td><td>01-08-24 to 26-11-24</td></tr>
<tr><td></td><td>05:30 pm to 07:00 pm</td><td>Fri</td><td>29-07-24 to 26-11-24</td></tr>
<tr><td>Section 3</td><td>02:30 pm to 04:00 pm</td><td>Fri</td><td>29-07-24 to 26-11-24</td></tr>
<tr><td></td><td>04:00 pm to 05:30 pm</td><td>Fri</td><td>29-07-24 to 26-11-24</td></tr>
<tr><td></td><td>05:30 pm to 07:00 pm</td><td>Fri</td><td>29-07-24 to 26-11-24</td></tr>
</table>
</td></tr>
<tr><td>Prerequisite</td><td>Not Applicable</td></tr>
<tr><td>Antirequisite</td><td>Not Applicable</td></tr>
<tr><td>Corequisite</td><td>Not Applicable</td></tr>
<tr><td>Course Description</td><td>The course begins with logical problem solving exercises and then introduces the students to the world of programming using Python. Further, the course includes rigorous exercises with scenario-based problem-solving questions and classical computer science puzzles. Python exercises including hands-on exercises to complete the given code, debugging code, writing alternate code and improvising code are included for various problems. The course also requires students to do a course project in a pair or a group of three. Moreover, the course introduces C language to the students for their future courses. This is a beginner-level computer programming course designed for engineering students.</td></tr>
</table>

Course Objectives	The course educational objectives (CEO) are to: • CEO1: Learn programming to solve logical problems and mathematical questions • CEO2: Learn programming to solve mathematical puzzles • CEO3: Learn Python programming to solve interdisciplinary problems • CEO4: Learn fundamentals of C
Learning Outcomes	After completing this course, a student should be able to, • CO1: Convert logical thinking into a pseudocode and solve mathematical questions by writing Python program. • CO2: Write user-defined functions and apply third party libraries in Python for problem solving • CO3: Understand, debug and improvise existing Python and C programs • CO4: Write C programs for problem-solving • CO5: Present project and document code concisely
Pedagogy	• The fundamental concepts teaching with hands-on Python programming exercises in which the students complete or improvise the given program. • Problem-solving exercises in which the students write code from scratch and write alternate code for the given problem. • Class quizzes to consolidate classroom learning. • Project-based learning with a Hackathon set up in which a team of two or three students develops one project and revise it with multiple iterations of the instructor's review to consolidate laboratory learnings. • Project presentation using videos and a structured brief report. • Use of LMS during mid-semester exam for a modern learning experience.
Expectation From Students	

Assessment/Evaluation	• Mid-Semester Examination:
	  ◦ Online Exam - 20%
	• End Semester Examination:
	  ◦ Written - 30%
	• Other Components:
	  ◦ Assignment - 20%
	  ◦ Project - 30%

Attendance Policy	As per Ahmedabad University Policy.

Project / Assignment Details	• Laboratory assignments contain three types of questions: coding for direct questions, coding for scenario-based descriptive problems, and coding for miscellaneous problems. • Mid-semester exam contains questions related to debugging code, completing code, predicting output of the code, and conceptual questions. • End-semester exam contains questions including writing code from scratch, conceptual questions, debugging and improvising code questions. • Project requires students to develop a working project using Python/C and other relevant technologies required. Students are required to brainstorm the project definition with the instructor to finalise the project objectives. Regular updates are required to be communicated. Final completion of the project with a concise structured abstract and a short video to present the project are required for viva.
Course Material	Text Book(s) • Programming in Python 3: A Complete Introduction to the Python Language, Mark Summerfield, 2nd Edition, Pearson, ISBN: ISBN: 978-93-528-6917-6, Year: Revised edition, 2022, Reference Book • Introduction to Computation and Programming Using Python: With Application to Computational Modeling and Understanding Data., John V. Guttag, 3rd Edition, MIT Press, ISBN: ISBN: 0262363437, 9780262363433., Year: 2021, • Programming in ANSI C, E. Balagurusamy, 8th Edition, Tata McGraw Hill, ISBN: ISBN-13: 978-9351343202, Year: 2019, • The Art of Computer Programming Volume 1, 2, 3, 4a, Donald E. Knuth, Addison-Wesley Professional, • Let Us C, Yashavant P. Kanetkar, Infinity Science Press, ISBN: ISBN: 1934015253, 9781934015254,
Additional Information	

Session Plan

NO.	TOPIC TITLE	TOPIC & SUBTOPIC DETAILS	READINGS,CASES,ETC.	ACTIVITIES	IMPORTANT DATES
1	Problem solving using Algorithm	Problem-solving using algorithms, flowcharts, pseudocode and programming languages	Course website: https://lms.ahduni.edu.in		
2	Lab session: Problem solving using Algorithm	Problem solving and pseudocode writing; Using RAPTOR for Flowcharts	Lab installation manual/video tutorials related to RAPTOR	Lab 1 - Client Developer interaction for software specification: Discussion-based team activity in which one student plays the role of a client and another student plays the role of a developer. They discuss the requirements and software specifications, critically identify exceptions, and prepare a flowchart for the solution.	
3	Programming Foundation	Language basics such as source code, object code, executable code, compilers and interpreters, Introduction to Python, structure of Python program	Handouts; Programming in Python 3 by Summerfield, Chapter 1	Activity based on code testing and debugging	

4	Lab session: Programming Foundation	Python IDE (IDLE/Spyder/Anaconda) installation, writing Hello World in Python, formatted input or output and basic programs in Python	Lab installation manual/video tutorials related to Python IDEs	Lab 2 - Logical debugging hands-on exercise to explore modern tools and understand their shortcomings: This lab session provides students with a comparison exercise with ChatGPT. Program codes will be given to the students and they will need to find whether the code is giving correct output or not, what are the logical errors, and whether the ChatGPT solutions be reliable or not. How to use AI tools in enhancing coding practices?	
5	Python programming identifiers	Constants and variables. Data types. Character set, tokens, keywords and identifiers.	Programming in Python 3 by Summerfield, Chapter 2	Identify and correct errors from a given program	
6	Lab session: Python programming identifiers	Writing code to solve mathematical questions using Python	Programming in Python 3 by Summerfield, Chapter 3	Lab 3 - Complete the code hands-on exercise: This lab session provides a hands-on exercise. The students will be given ready-to-run Python codes with a few blank lines. The students will be expected to complete the missing code lines and produce the expected output. The problems will be based on Mathematics.	
7	Operators	Python operators: arithmetic, assignment, logical, comparison, membership, and bitwise. Precedence and associativity among operators. Type conversion.	Handouts	Activity based on Python Operators	

8	Lab session: Operators	Writing code to implement solution for various condition-based problems	Handouts	Lab 4 - Improve the code exercise: This lab session will provide students with an activity to write minimal code. Students will be given a sample code for a problem. They need to come up with creative solutions, including different ways of solving and solutions with less number of code lines.	
9	Control statements	If, If..else, and Nested if. Real-world examples, brainstorming exercise	Programming in Python 3 by Summerfield, Chapter 4	Activity based on Python decision making statements	
10	Lab session: Control statements	Implementing activities based on decision making statements	Programming in Python 3 by Summerfield, Chapter 4	Lab 5 - Logical programming exercise: This lab session will provide students with an activity to write logical code. The student will be given a problem statements which they will be required to convert into a program. They will need to divide the problem into logical steps and convert the steps into Python code.	
11	Decision making and Looping	Looping statements: for and while loop statements	Programming in Python 3 by Summerfield, Chapter 4	Generating a pattern using loops	

12	Lab session: Decision making and Looping	Implement solution for repetitive or loop-based problems including patterns		Lab 6 - Programming beyond question-answers exercise: This exercise will provide students to relate programming and art together. Students will be asked to draw various shapes, such as pyramids, a butterfly, a diamond and a kite using code.	
13	Functions	Library and User-defined functions in Python	Programming in Python 3 by Summerfield, Chapter 5	Activity based on user-defined functions	
14	Lab session: Functions	Writing code to create user-defined functions for the given problems	Programming in Python 3 by Summerfield, Chapter 6	Lab 7 - Code reusability exercise: This lab session will provide students with an exercise of code reusability. The students will be given a problem in which they need to perform a task multiple times. They will be taught the modular programming way of writing a user-defined function that can do the job without the repetition of the same code.	
15	Mid semester examination				
16	File management	File management in Python; Read Write Append Operations	Programming in Python 3 by Summerfield, Chapter 7	Project discussion	

17	Lab Session: Projects that bring value	File handling exercises using Python	Programming in Python 3 by Summerfield, Chapter 8	Lab 8 - Interdisciplinary project demos and definition discussion: This lab session will provide examples and brainstorming discussions of projects that value the most in the society, address bigger challenges of the society, links theory and real-world problems, and interdisciplinary problems.	
18	Python collections and Libraries	Python collection data types/array - List, Set, Tuple, Dictionary; Using third parties libraries - Numpy, Pandas, etc.	Programming in Python 3 by Summerfield, Chapter 3	-	
19	Lab session: Python collections and Libraries	Python exercises using list, set, tuple and dictionary	Programming in Python 3 by Summerfield, Chapter 4	Lab 9 - Programming and need of data structures: This lab activity will be focused on a systematic way of dealing with multiple types of data. The students will be given a task to store the data of a telephone directory and retrieve the details of a particular person.	
20	Introduction to C	C Programming Basics; Structure of C program	Programming in ANSI C by E Balagurusamy, 8th Edi. Chapter 2, 3	Activity - Tracing a program	
21	Lab session: Introduction to C	Installation of DevC++, introduction to C and basic exercises	Lab installation video tutorials related to C IDEs	Lab 10 - Converting one type of programming syntax into another exercise: This lab session will provide a code transfer exercise to convert Python code into C code.	

22	Decision making and looping in C	if, if-else, switch, while, do-while, for statements	Programming in ANSI C - Chapter 6, 7	Activity based on improving a given code	
23	Lab session: Decision making and looping in C	Condition statements and looping programs in C	Programming in ANSI C - Chapter 6, 8	Lab 11 - Finding and fixing the errors exercise: This lab session will provide an exercise to find errors in the given code. The students will be given a code, and they will need to identify all the errors and fix them.	
24	Array and String	Array, pointers and string processing in C	Programming in ANSI C - Chapter 8, 9	-	
25	Lab session: Array and String	C array, pointers and string exercises	Programming in ANSI C - Chapter 8, 10	Lab 12 - Extending the code exercise: This lab session will be focused on extending the code. The students will be given a code and then they need to extend the code with additional features. Common interview questions will be considered for this exercise.	
26	User defined functions	Library and User-defined functions in C, Recursion	Programming in ANSI C - Chapter 10	Convert given code into efficient C code	
27	Lab session: User defined functions and C File Management	C user-defined function, File management operations exercises	Programming in ANSI C - Chapter 11	Lab 13 - Discussion to code exercise: This lab session will be focused on classical problems in computer science. An animated solution of the problem will be shown to the students which they need to convert into a program.	

28	Reviews and Reflections	-			
29	Reviews and Reflections	-			
30	End semester examination				