

20IT4353-ADVANCED PROGRAMMING LAB-I

Course Category:	Programme Core							Credits:					1.5		
Course Type:	Lab							Lecture-Tutorial-Practice:					0-0-3		
Prerequisites:	20ES1103 Programming for Problem Solving 20ES2103 Object Oriented Programming using Python							Continuous Evaluation:					30		
								Semester End Evaluation:					70		
								Total Marks:					100		
Course Outcomes	Upon successful completion of the course, the student will be able to:														
	CO1	Demonstrate the knowledge of problem solving and to find solutions that use different types of programming paradigms.													
	CO2	Apply the knowledge of number theory to solve problems and generatesolutions													
	CO3	Design solutions to the problems by applying linear and non-linear data structures													
	CO4	Develop combinatory solutions to the real world problems													
	CO5	Execute basic algorithmic ideas using greedy approach to solve competitive programming problems													
	CO6	Analyze dynamic programming approaches to generate solution to the problems													
Contribution of Course Outcomes towards achievement of Program Outcomes (L-Low, Medium-M, H- High)		PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
	CO1	M	M	M			M					H	H	M	H
	CO2	L	M	M			M					M	M	L	L
	CO3	H	M	H			M					H	H	H	H
	CO4	L	M	M			M					M	M	L	L
	CO5	H	M	H			H					H	H	H	H
	CO6	H	M	H			H					H	H	H	H
Course Content	Week 1: Understand and identify the time complexity of a real world problem a. Identify the time complexity of loops and write it in asymptotic notations b. Solve the real world array problems and find their time complexities Week 2& 3: Design Solutions using searching and sorting algorithms a. Solve programs from different coding platforms to make use of searching and sorting algorithms														

	<p>Week 4: Derive solutions to problems that make use of Graph algorithms</p> <ul style="list-style-type: none"> a. Design and develop programs using Depth and breadth first search algorithms b. Identify the solutions using Warshalls and Bellman Ford's algorithms <p>Week 5, 6 & 7: Identify the need and importance in the use of Greedy and dynamic algorithms in problem solving</p> <ul style="list-style-type: none"> a. Apply greedy technique to find the solutions to real world problems <p>Week 8: Programs on the implementation of methods and operations of data structures of Python</p> <ul style="list-style-type: none"> a. Practice all the methods of all the data structures from python <p>Week 9 & 10: Implement programs to solve the problems using String manipulation and string matching algorithms</p> <ul style="list-style-type: none"> a. Design solutions by make use of string manipulation and matching algorithms <p>Week 11 & 12: Solve programming problems based on math and combinatorics</p> <ul style="list-style-type: none"> a. Modular arithmetic b. Modular exponentiation and multiplicative inverse c. Greatest common Divisor d. Mike and Matrix Game e. Sum of Series and other problems
Text books and Reference books	<p>Text Book(s):</p> <ul style="list-style-type: none"> [1]. Halim, Steven and Halim, Felix, Competitive Programming 1, 2013 [2]. Reema Thareja, "Python Programming Using Problem Solving Approach", Oxford University Press, 2019. <p>Reference Books:</p> <ul style="list-style-type: none"> [1]. Antti Laaksonen, "Guide to Competitive Programming", 1st edition, Springer International Publishing, 2017 [2]. Ahmed Shamsul Arefin, Art of Programming Contest, ACM Solver, Second Edition, 2012 [3]. Zed Shah, "Learn Python The Hard Way", Third edition, Addison-Wesley, 2013. [4]. John V. Guttag, "Introduction to Computation and Programming Using Python", The MIT Press, 2013
E-resources and other digital material	<ul style="list-style-type: none"> [1]. Filipp Rukhovich, Competitive Programming for beginners, [COURSERA]. (19-05-2021), Available: https://www.coursera.org/learn/competitive-programming-for-beginners [2]. Prof. Neeldhara, IIT Gandhinagar, Getting Started with Competitive Programming, [NPTEL], (19-05-2021), Available: https://onlinecourses.nptel.ac.in/noc21_cs99/preview [3]. Prof. Erik Demaine, Prof. Ronald Rivest, Prof. Srinivas Devas MIT Open Courseware, Introduction to Algorithms, Getting Started with Competitive Programming, [MIT],

	<p>(19-05-2021),Available:https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-006-introduction-to-algorithms-spring-2008/index.htm</p> <p>[4].Hacker Rank, 19-05-2021 Available https://www.hackerrank.com/</p> <p>[5].Leet Code, 19-05-2021 Availablehttps://leetcode.com/</p> <p>[6].Hacker Earth, 19-05-2021 Available https://www.hackerearth.com/</p> <p>[7].Topcoder, 19-05-2021 Available https://www.topcoder.com/challenges/</p> <p>[8].Coder Byte, 19-05-2021 Available https://www.coderbyte.com/</p> <p>[9].Code wars, 19-05-2021 Available https://www.codewars.com/</p> <p>[10].Code Signals, 19-05-2021 Available https://codesignal.com/</p> <p>[11].Code Chef, 19-05-2021 Available https://www.codechef.com/</p>
--	--