	National Institute of Technology Meghalaya An Institute of National Importance	CURRICULUM

Programme	**Bachelor of Technology in Computer Science and Engineering**	Year of Regulation	**2019-20**
Department	**Computer Science and Engineering**	Semester	**IV**

Course Code	Course Name	Credit Structure				Marks Distribution			
		L	T	P	C	INT	MID	END	Total
CS212	**Analysis and Design of Algorithms**	3	0	0	3	50	50	100	200

Course Objectives	To teach paradigms and approaches used to analyze and design algorithms and to appreciate the impact of algorithm design in practice.	Course Outcomes	CO1	Analyze the asymptotic performance of algorithms.
	To make students understand how asymptotic notation is used to provide a rough classification of algorithms.		CO2	Write rigorous correctness proofs for algorithms.
	To explain different computational models and various complexity measures to analyze the complexity/performance of different algorithms.		CO3	Apply important algorithmic design paradigms and methods of analysis.
	To teach various advanced design and analysis techniques such as greedy algorithms, dynamic programming.		CO4	Synthesize efficient algorithms in common engineering design situations.
	Know the concepts of tractable and intractable problems and the classes P, NP and NP-complete problems.			

No.	COs	Mapping with Program Outcomes (POs)												Mapping with PSOs		
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
1	CO1	2	1	1	0	1	0	0	0	0	1	1	1	1	0	1
2	CO2	2	1	1	1	1	1	0	0	1	0	0	0	1	1	1
3	CO3	1	1	1	1	1	0	0	0	1	0	0	0	2	2	0
4	CO4	2	2	2	2	0	0	0	0	0	0	0	0	1	1	1

SYLLABUS			
No.	Content	Hours	Cos
I	**Introduction** Algorithm Specification, Algorithm Analysis, Analysis of Recursive Algorithms.	06	CO1 CO2
II	**Sorting and Selection** Brute Force Approaches- Sequential Search, Bubble Sort, Selection Sort, Exhaustive Searching, Divide-and-Conquer Approach – Merge Sort, Quick Sort, Closest-pair Problem, Convex Hull Problem, Decrease-and-Conquer Approach – Insertion Sort, Topological Sort, Linear Sorting – Counting Sort, Bucket Sort, Radix Sort	09	CO2 CO3
III	**Greedy Algorithms** Introduction, Knapsack Problem, Optimal Tree Problems – Optimal Merge, Huffman Coding; Optimal Graph Problems – Minimum Spanning Trees, Single-source Shortest-Path; Scheduling Problems – Scheduling without deadline, Scheduling with deadline	07	CO2 CO3
IV	**Dynamic Programming** Basics of Dynamic Programming, Fibonacci Problem, Multistage Graph Problem, All Pairs Shortest-path Algorithm, Travelling Salesman Problem, Chain Matrix Multiplication, Knapsack Problem, Optimal Binary Search Trees,	08	CO2 CO3
V	**String processing** String searching and Pattern matching, Knuth-Morris-Pratt algorithm and its analysis.	04	CO2 CO4
VI	**Computational Complexity Classes** Upper and Lower Bound Theory, Class P, NP Class, NP- Complete	02	CO1 CO4
	Total Hours	36	

Essential Readings

1. A. Aho, J. Hopcroft and J. Ullman, "The Design and Analysis of Computer Algorithms", 4th Impression, Addison-Wesley, 2009.

2. E Horowitz, S Sahni, and S Rajasekhran, "Fundamentals of Computer Algorithms", 2nd Edition, Universities Press, 2008.

3. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, "Introduction to Algorithms", 3rd Edition, Pearson, 2010.

4. S. Sridhar, "Design and Analysis of Algorithms", 1st Edition, Oxford University Press, 2015.

Supplementary Readings

1. J. Kleinberg, E Tardos, "Algorithm Design", 1st Edition, Pearson, 2014.

2. S. Dasgupta, C. H. Papadimitriou, and U. V. Vazirani, "Algorithms", 2nd Edition, Tata McGraw Hill, 2016.

3. Steven S Skiena, "The Algorithm Design Manual", 2nd Edition, Springer, 2011.

4. H Bashin, "Algorithms Design and Analysis", 1st Edition, Oxford University Press, 2015.