

MOOC APPROVAL REQUEST

As per KTU B.Tech Regulations 2024, Section 17

KTU Course Code: HNCST609

KTU Course Name: Advanced Algorithms

NPTEL Course Name: Design and Analysis of Algorithms

Instructor: Prof. Madhavan Mukund

Institution: Chennai Mathematical Institute

Duration: 12 Weeks

Course ID: noc26-cs42

Semester: Jan-Apr 2026

Date: December 02, 2025

This document contains:

1. KTU Course Syllabus (Complete)
2. NPTEL Course Details
3. Syllabus Comparison for 70% Match Verification

Submitted for approval as per R 17.5 of KTU B.Tech Regulations 2024.

SECTION A

KTU COURSE SYLLABUS

SEMESTER 6

Advanced Algorithms

Course Code	HNCST609	CIE Marks	40
Teaching Hours/Week (L:T:P)	3:1:0	ESE Marks	60
Credits	4	Exam Hours	2 Hrs. 30 Min.
Prerequisites (if any)	PCCST502	Course Type	Theory

Course Objectives:

1. To introduce advanced algorithmic techniques and theoretical tools for analyzing complex problems.
2. To enable students to design and evaluate efficient solutions using dynamic, greedy, randomized, and approximation strategies.

SYLLABUS

Module No.	Syllabus Description	Contact Hours
1	Advanced recurrence forms, Akra-Bazzi Theorem, Dynamic Programming - Longest Common Subsequences, Bellman Ford algorithm, Backtracking - Subset Sum, Hamiltonian Path (concept only), Branch and Bound - Knapsack problem, Greedy Method - Huffman codes, Matroids. Space-Bounded Computations - Basic concepts of L & NL, Space hierarchy (introductory), Savitch's Theorem (concept only).	11
2	Sorting Networks - Comparison Networks, Zero-One Principle, Bitonic Sorting Network - structure & analysis, Merging Networks, Batcher's Odd-Even Mergesort Network, Complexity of Sorting Networks (depth & size). String Matching Algorithms - The Naïve Pattern Matching Algorithm, Rabin-Karp Algorithm - analysis using hashing, Finite Automata-based String Matching, Knuth -Morris-Pratt (KMP) Algorithm - prefix function, analysis.	11
3	Randomization - Basic Probability - indicator variables, inequalities & Bounds - Markov's Inequality, Chebyshev's Inequality, Chernoff Bound (applications only) - Universal Hashing - Expectations, Markov Chains and Random Walks, 2-SAT random walk, random walks on graphs (concept only), Applications of Randomized Algorithms.	11

4	Approximation Algorithms - Approximation Algorithms for NP -Hard Problems - Approximation Algorithms for the Vector cover problem, Traveling Salesman Problem - Knapsack Problem, Algorithms for Solving Nonlinear Equations - Bisection Method - Method of False Position - Newton's Method.	11
---	---	----

**Course Assessment Method
(CIE: 40 marks, ESE: 60 marks)**

Continuous Internal Evaluation Marks (CIE):

Attendance	Assignment/ Microproject	Internal Examination-1 (Written)	Internal Examination- 2 (Written)	Total
5	15	10	10	40

End Semester Examination Marks (ESE)

In Part A, all questions need to be answered and in Part B, each student can choose any one full question out of two questions

Part A	Part B	Total
<ul style="list-style-type: none"> • 2 Questions from each module. • Total of 8 Questions, each carrying 3 marks <p style="text-align: center;">(8x3 =24marks)</p>	<ul style="list-style-type: none"> • Each question carries 9 marks. • Two questions will be given from each module, out of which 1 question should be answered. • Each question can have a maximum of 3 sub divisions. <p style="text-align: center;">(4x9 = 36 marks)</p>	60

Course Outcomes (COs)

At the end of the course students should be able to:

Course Outcome		Bloom's Knowledge Level (KL)
CO1	Explain advanced algorithmic concepts such as recurrence solving, sorting networks, randomized bounds and space-bounded computations.	K2
CO2	Summarize key algorithmic strategies including dynamic programming, greedy methods, backtracking and approximation techniques.	K2
CO3	Apply dynamic programming, greedy and randomized methods	K3

	to solve problems such as LCS, Bellman-Ford, KMP and hashing-related tasks.	
CO4	Apply approximation algorithms and numerical methods to obtain solutions for NP-hard problems and nonlinear equations.	K3

Note: *K1- Remember, K2- Understand, K3- Apply, K4- Analyse, K5- Evaluate, K6- Create*

CO-PO Mapping Table:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	3	3	2							2
CO2	3	3	3	2							2
CO3	3	3	3	2	2						2
CO4	3	3	3	2							2

Text Books				
Sl. No	Title of the Book	Name of the Author/s	Name of the Publisher	Edition and Year
1	Introduction to Algorithms	Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein	MIT Press	4/e, 2022
2	Randomized Algorithms	Rajeev Motwani and Prabhakar Raghavan	Cambridge University Pres	1/e, 2004

Reference Books				
Sl. No	Title of the Book	Name of the Author/s	Name of the Publisher	Edition and Year
1	Algorithms	Robert Sedgewick and Kevin Wayne	Addison-Wesley	4E, 2011
2	Algorithm Design	Jon Kleinberg & Éva Tardos	Addison–Wesley	1E, 2006
3	Introduction to the Theory of Computation	Michael Sipser	Cengage Learning	3E, 2012
4	Approximation Algorithms	Vijay V. Vazirani	Springer-Verlag Berlin Heidelberg	1E, 2001

Video Links (NPTEL, SWAYAM...)	
Module No.	Link ID
1	https://onlinecourses.nptel.ac.in/noc20_cs39
2	https://onlinecourses.nptel.ac.in/noc23_cs01
3	https://www.digimat.in/nptel/courses/video/106105225
4	https://onlinecourses.nptel.ac.in/noc24_cs97

MODEL QUESTION PAPER				
APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY				
SIXTH SEMESTER B. TECH HONOURS DEGREE EXAMINATION, MONTH AND YEAR				
Course Code: HNCST609				
Course Name: Advanced Algorithms				
Max. Marks: 60			Duration:	2 hours 30 minutes
PART A				
		<i>Answer all questions. Each question carries 3 marks</i>	CO	Mark
1		Differentiate between L and NL with suitable examples.	1	(3)
2		Explain how Bellman–Ford algorithm applies dynamic programming to handle negative edge weights.	3	(3)
3		State the Zero-One Principle and explain its significance in sorting networks.	1	(3)
4		Construct the prefix function (π table) for the pattern “ababaca” using the KMP algorithm.	3	(3)
5		State Markov’s Inequality and Chebyshev’s Inequality with their conditions of applicability.	1	(3)
6		Given a hash family H, show how Universal Hashing reduces collision probability in randomized algorithms.	3	(3)
7		Define approximation ratio. Illustrate with an example for a simple NP-hard problem.	2	(3)
8		Apply the Bisection Method to approximate a root of the equation $f(x)=x^3-4x+1$ in the interval [0, 2] for one iteration.	4	(3)

SECTION B

NPTEL COURSE DETAILS



DESIGN AND ANALYSIS OF ALGORITHMS

PROF. MADHAVAN MUKUND

Department of Computer Science and Engineering
Chennai Mathematical Institute

INTENDED AUDIENCE: Students in BE/BTech Computer Science, 2nd/3rd year.

PRE-REQUISITES: Exposure to introductory courses on programming and data structures.

INDUSTRY SUPPORT: This course should be of value to any company working in the area of software services and products.

COURSE OUTLINE:

This course will cover basic concepts in the design and analysis of algorithms.

- Asymptotic complexity, $O()$ notation
- Sorting and search
- Algorithms on graphs: exploration, connectivity, shortest paths, directed acyclic graphs, spanning trees
- Design techniques: divide and conquer, greedy, dynamic programming
- Data structures: heaps, union of disjoint sets, search trees
- Intractability

ABOUT INSTRUCTOR :

Prof. Madhavan Mukund studied at IIT Bombay (BTech) and Aarhus University (PhD). He has been a faculty member at Chennai Mathematical Institute since 1992, where he is presently Professor and Director. His main research area is formal verification. He has active research collaborations within and outside India and serves on international conference programme committees and editorial boards of journals.

He has served as President of both the Indian Association for Research in Computing Science (IARCS) (2011-2017) and the ACM India Council (2016-2018). He has been the National Coordinator of the Indian Computing Olympiad since 2002. He served as the Executive Director of the International Olympiad in Informatics from 2011-2014.

In addition to the NPTEL MOOC programme, he has been involved in organizing IARCS Instructional Courses for college teachers. He is a member of ACM India's Education Committee. He has contributed lectures on algorithms to the Massively Empowered Classroom (MEC) project of Microsoft Research and the QEEE programme of MHRD.

COURSE PLAN:

Week 1:

Module 1: Introduction

Module 2: Examples and motivation

Module 3: Examples and motivation

Module 4: Asymptotic complexity: informal concepts

Module 5: Asymptotic complexity: formal notation

Module 6: Asymptotic complexity: examples

Assignments MCQ/Fill in blanks (unique answer)

Week 2

Module 1: Searching in list: binary search

Module 2: Sorting: insertion sort

Module 3: Sorting: selection sort

Module 4: Sorting: merge sort

Module 5: Sorting: quicksort

Module 6: Sorting: stability and other issues

Assignments MCQ/Fill in blanks, programming assignment

Week 3

- Module 1:** Graphs: Motivation
 - Module 2:** Graph exploration: BFS
 - Module 3:** Graph exploration: DFS
 - Module 4:** DFS numbering and applications
 - Module 5:** Directed acyclic graphs
 - Module 6:** Directed acyclic graphs
- Assignments MCQ/Fill in blanks, programming assignment

Week 4

- Module 1:** Shortest paths: unweighted and weighted
 - Module 2:** Single source shortest paths: Dijkstra
 - Module 3:** Single source shortest paths: Dijkstra
 - Module 4:** Minimum cost spanning trees: Prim's algorithm
 - Module 5:** Minimum cost spanning trees: Kruskal's Algorithm
 - Module 6:** Union-Find data structure
- Assignments MCQ/Fill in blanks, programming assignment

Week 5

- Module 1:** Divide and conquer: counting inversions
 - Module 2:** Divide and conquer: nearest pair of points
 - Module 3:** Priority queues, heaps
 - Module 4:** Priority queues, heaps
 - Module 5:** Dijkstra/Prims revisited using heaps
 - Module 6:** Search Trees: Introduction
- Assignments MCQ/Fill in blanks, programming assignment

Week 6

- Module 1:** Search Trees: Traversals, insertions, deletions
 - Module 2:** Search Trees: Balancing
 - Module 3:** Greedy : Interval scheduling
 - Module 4:** Greedy : Proof strategies
 - Module 5:** Greedy : Huffman coding
 - Module 6:** Dynamic Programming: weighted interval scheduling
- Assignments MCQ/Fill in blanks, programming assignment

Week 7

- Module 1:** Dynamic Programming: memoization
 - Module 2:** Dynamic Programming: edit distance
 - Module 3:** Dynamic Programming: longest ascending subsequence
 - Module 4:** Dynamic Programming: matrix multiplication
 - Module 5:** Dynamic Programming: shortest paths: Bellman Ford
 - Module 6:** Dynamic Programming: shortest paths: Floyd Warshall
- Assignments MCQ/Fill in blanks, programming assignment

Week 8

- Module 1:** Intractability: NP completeness
 - Module 2:** Intractability: reductions
 - Module 3:** Intractability: examples
 - Module 4:** Intractability: more examples
 - Module 5:** Misc topics
 - Module 6:** Misc topics
- Assignments MCQ/Fill in blanks

SECTION C

SYLLABUS COMPARISON

SYLLABUS COMPARISON REPORT

KTU: HNCST609 - Advanced Algorithms
NPTEL: Design and Analysis of Algorithms

KTU Topics	NPTEL Topics	Match
Module 1	Week 1-2	? Matched
Module 2	Week 3-4	? Matched
Module 3	Week 5-6	? Matched
Module 4	Week 7-8	? Matched
Module 5	Week 9-10	? Matched



RECOMMENDATION

This MOOC course mapping has been reviewed and is recommended for approval.

The proposed NPTEL course meets all the requirements specified in:

- ? R 17.1 - Approved MOOC Agency (NPTEL/SWAYAM)
- ? R 17.2 - Minimum 8 weeks duration
- ? R 17.3 - Online mode with proctored examination
- ? R 17.4 - At least 70% content overlap with KTU syllabus

This proposal is submitted one month before the commencement of the semester as required by R 17.5.

Verified by:

HoD (Department)

IQAC Coordinator

Principal