

8. Combinatorial Optimization
 - a. Knapsack and Bin Packing
 - b. Travelling Salesperson Problem
 - c. Graph Cover and Colouring Problems
 - d. Scheduling Problems
9. NP-Completeness, Approximation, Randomization
 - a. Classes P and NP
 - b. NP-Completeness and Reducibility
 - c. NP-Completeness Proofs
 - d. Polynomial Approximation Schemes
 - e. Feasibility and Hardness of Approximation
 - f. Basics of Randomized Algorithms

Course Learning Strategy

Algorithm design, especially the starting steps, is learnt by a combination of reading and studying good algorithms, developing your own algorithms and correcting or improving solutions developed by others, including peers in a group learning effort. All the three are important to achieve success. In addition, it is important to be able to code some alternative algorithms into programs, execute them and compare their performances to check whether theoretical analysis confirms with experimentation. A text-book must be available for study. It is preferred to generally follow a single text-book. However, referring to other text-books at times to understand different topics may be preferred. While studying in a group, it is useful if individual members follow some other books so that a variety of inputs are obtained. Web and video resources are good additional inputs. The steps include reading the chapters relevant, watching the video, discussing the concepts in a group and then solving a set of problems. The solutions to the problems solved by the group can be discussed together and one or more final versions may be accepted. Trying to check, other member's solutions are a very important aspect of learning algorithms design.

Duration and Venue

Duration : The duration of the workshop is 6 working days. It will start on Monday 25th May, 2015 at 9:00 AM and will end on Saturday 30th May, 2015. The participants must report to the respective remote centres by 8:00 AM on 25th May, 2015.

Venue: 221 remote centers located in different parts of the country. The list of participating remote centers are given along with online application form.

Teaching Faculty

Prof. Partha Pratim Chakrabarti, Department of Computer Science and Engineering, IIT Kharagpur,
email: ppchak@cse.iitkgp.ernet.in.

Prof. Partha Pratim Das, Department of Computer Science and Engineering, IIT Kharagpur,
email: ppd@cse.iitkgp.ernet.in.

Prof. Pallab Dasgupta, Department of Computer Science and Engineering, IIT Kharagpur,
email: pallab@cse.iitkgp.ernet.in

Eligibility

The participants must be a regular / visiting faculty in Engineering Degree College / Polytechnic / Other degree colleges. They must have at least B. Tech degree or M.Sc. degree (CS, IT, Mathematics) or MCA or equivalent degree. They should be teaching in departments (Electrical, Electronics, Information Technology, Computer Science / Engineering, Mathematics & Computer Application), where subjects like Algorithm Design, Programming & Data Structures, Numerical Algorithms, Parallel Algorithms, Distributed Algorithms, Software Engineering, Object-Oriented Analysis & Design etc. are taught.

Who may benefit

The workshop is likely to benefit regular/visiting faculty colleagues who are teaching subjects like Algorithm Design, Programming & Data Structures, Numerical Algorithms, Parallel Algorithms, Distributed Algorithms, Software Engineering, Object-Oriented Analysis & Design at the undergraduate or the postgraduate level.

Note

Please note that this ISTE STTP is conducted under CEP IIT Kharagpur. Live recording of the course and other created contents will be released under Open Source through a portal. The recorded CD/DVD of the course lectures will be available for distribution, at cost, to any individual or institution. All participants are required to sign an undertaking for such release of contents contributed by them during and after the workshop. The recognition and citation will naturally be made for all contributors.

Course Fee

Since the workshop is funded by the National Mission on Education through ICT (MHRD, Government of India), there is no course fee for participation.

Accommodation and other support for outstation participants

Remote Centers are being funded to provide tea/lunch on each day of the workshop, and for accommodation, wherever available, for a limited number of outstation participants. **Travel expenses up to Rs. 1000/- one way and one-time will be reimbursed against proof of actual expenditure, for participants beyond a distance of 100 Km from the Remote Centre.**

How to Apply

Those wishing to attend this course should register online <http://www.nmeict.iitkgp.ernet.in/algorithmmain.php>
Online registration opens on 25th March, 2015

Address for Communication

Admin Team, Project "T10KT", IIT Kharagpur
Vikramshila Building, Ground floor, Kalidas Auditorium
IIT Kharagpur, Kharagpur-721302
Tel: Admin Team : +91 3222-281497
Account Team : +91 3222-281498
Moodle Team : +91 3222-281070
A-View Team : +91 3222-281072
Mobile : +91 8145226903
email: office_nmeict@iitkgp.ac.in
officenmeict@gmail.com

TWO WEEK ISTE STTP ON INTRODUCTION TO DESIGN OF ALGORITHMS

National Mission on Education through ICT
(MHRD, Govt. of India)

May 25–30, 2015



**Indian Institute of Technology
Kharagpur 721302
India**

Introduction

An important initiative has been taken by IIT Bombay and IIT Kharagpur to work with Engineering Colleges of India to enhance the teaching skills of our faculty colleagues in core Engineering and Science subjects under Train Ten Thousand Teachers programme. Participating teachers attend live lectures at a remote center close to their own college, and also attend tutorial and lab sessions conducted in the same centers. The lecture transmission and live interaction takes place in distance mode using A-VIEW technology through internet at the selected remote centers across the country. This initiative is a part of the 'National Mission on Education through ICT,' which is supported by MHRD. Faculty coordinators are appointed at each remote centre to handle the technology infrastructure and other operational logistics. Additionally, for each workshop there will be a Faculty assigned as Workshop Coordinator for that subject, who will help in conducting laboratories and tutorials at each center.

We invite expert faculty members from various remote centers to a five-day ISTE STTP for Coordinators held at IIT Kharagpur or at IIT Bombay, at least two months before the Main Workshop. The trained Coordinators then act as Workshop Coordinators during the Main Workshop, liaising between the participants at their Remote Centers and IIT Kharagpur / IIT Bombay from where the lecture is transmitted live. During the Main Workshop, the Workshop Coordinator at every center supervises the tutorials and laboratories. All the lectures and tutorial sessions are recorded at IIT Kharagpur or at IIT Bombay. The final edited audio-visual contents, along with other course material are released under Open Source. The contents can be freely used later by all teachers and students.

Since December 2009, a number of two-week ISTE workshops were conducted on different subjects, namely "Effective teaching/ learning of Computer Programming," "Database Management Systems," "Basic Electronics," "Thermodynamics," "Software Development Techniques for Teachers of Engineering and Science Colleges," "Heat Transfer", "Solar Photovoltaics", "Introduction to Research Methodology", "Engineering Thermodynamics", "Analog Electronics", "Research Methods in Education Technology", "Signal & Systems", "Fluid Mechanics" and "Control Systems". We have reached more than **50,000** teachers and helped them to enhance their teaching skills at around **345** distinct Remote Centers across the country.

In the backdrop of the success of these workshops, we now announce another 6 day ISTE STTP on **Introduction to Design of Algorithms** during **May 25, 2015 to May 30, 2015** under **Blended MOOCs** (Massive Open Online Courses) model. Here,

1. The participating teachers will complete the equivalent of two-week full time work online, spread over 6 physical weeks where video lectures of the Co-ordinators' STTP and assignments will be uploaded beforehand.
2. Participants will then assemble at the selected Remote Centers for 6 days face to face interaction and lecture sessions through A-VIEW and will complete team assignments, tutorials, quizzes etc.

3. Offline assignments will be uploaded and the participants will have to complete these assignments within a stipulated time.

The above proposed model is tentative and subject to minor changes.

Course Justification

Design and Analysis of Algorithms is fundamental to correct and efficient design of software programs. Every graduate in Computer Science or Computer Science and Engineering needs to have a detailed exposure to (a) Elements of algorithm analysis, (b) Basic techniques of design of sequential algorithms and (c) Important algorithms that are widely used in applications.

Course Overview

Algorithm Design is a core course for any undergraduate programme in Computer Science or Computer Science and Engineering. It is expected to be the first course on this topic and is typically meant for students who already have some introductory knowledge of programming and data structures.

The course shall cover the basic foundations of designing correct and efficient sequential algorithm - through a process of mathematical analysis and logical design steps - which can be translated to software programs for effective deployment in practice. Analysis involves an understanding of algorithm complexity through asymptotic analysis of time and space requirements under worst, average and amortized scenarios. Design involves developing and refining an initial solution of the problem using fundamental design techniques like divide-and-conquer, dynamic programming, greedy decomposition, backtracking, branch-and-bound and data structuring. The techniques will be applied to develop good quality algorithms for important domains like sorting, searching, strings, graphs, matrices and optimization. The course shall also introduce the notion of hard problems through the theory of NP-Completeness and the development of approximation algorithms to provide tractable solutions to hard problems. The course is restricted to sequential algorithms and does not address parallel and distributed algorithms. Through this course a student is expected to be able to learn to properly define a problem, examine its inherent complexity and explore different ways to develop an algorithmic solution by analyzing the various alternatives to finally develop a good practical solution that is supported by logical and theoretical justification.

Course Objective

Students pursuing this course should be able to

1. Given an English language problem description, define the problem precisely with input/output requirements, examine its inherent complexity and develop a generic or set of initial solutions (which can be explored for various design options) and justify their correctness.
2. Given an algorithm description, analyze the time and space complexity of the algorithm in the worst case, average case and amortized scenario as needed in terms of asymptotic orders of complexity.

3. Given a problem definition, explore different alternative algorithmic solutions, compare them with respect to time and space complexity and choose the design schemes and/or design parameters and data structures appropriately to obtain the best possible choice(s) that can be converted to an executable program.
4. Design and analyze algorithms using the methods studied to solve problems in important applications including those related to sorting, searching, strings, graphs, matrices, data structuring and combinatorial optimization.
5. Examine and prove whether a problem is of polynomial complexity, hard (NP-Complete) or otherwise and develop optimal and approximation algorithms for them as applicable.

Course Modules

1. Elements of Algorithm Analysis
 - a. Problems, Solutions, Algorithms
 - b. Overall Process of Algorithm Design
 - c. Efficient Algorithms and Hard Problems
2. Elements of Algorithm Analysis
 - a. Time and Space Complexity
 - b. Asymptotic Growth and Orders of Complexity
 - c. Complexity Analysis using Recurrence Equations
 - d. Master Theorem
3. Basic Design Methods
 - a. Divide-and-Conquer
 - b. Greedy Decomposition
 - c. Dynamic Programming
 - d. Backtracking
 - e. Branch-and-Bound
 - f. Role of Data Structures
4. Data Structuring
 - a. Data and Operations
 - b. Stacks and Queues
 - c. Basic Binary Search Tree
 - d. Balanced Search Trees
 - e. Hash Tables
5. Searching and Sorting
 - a. Linear and Binary Search
 - b. Mergesort, Quicksort
 - c. Insertion and Heapsort
 - d. Linear Time Selection and Median Finding
6. Basic Graph Algorithms
 - a. Paths and Connected Components
 - b. Bi-Connected Components
 - c. Minimal Cost Spanning Tree
 - d. Dijkstra's Shortest Path Algorithm
 - e. Max-Flow
7. Matrix and String Problems
 - a. Matrix Multiplication
 - b. Generalized Transitive Closure of a Graph
 - c. String Matching