Introduction

CS31005: Algorithms-II

Autumn 2020

IIT Kharagpur

Instructors and Class Timings

- Instructors
  - Prof. Abhijit Das
    - Email: abhij@cse.iitkgp.ac.in
  - Prof. Arobinda Gupta
    - Email: agupta@cse.iitkgp.ac.in
- Class Timings
  - Lecture hours: Tue 02:00pm–04:00pm [Slot U4], Wed 08:00pm–09:30pm [Slot U(A)]
  - Doubt-clearance and tutorial sessions: Mon 03:00pm–05:00pm [Slot U4]
- Course webpage:
  https://cse.iitkgp.ac.in/~abhij/course/theory/Algo2/Autumn20

Mode of Conduction

- This is an online class because of the current extraordinary situation. The class will be conducted in the following mode:
  - Every week, by Sunday night, we will upload around 3 to 4 hours of recorded lectures in an youtube channel (common for both sections). The lectures will be linked from the appropriate topics in the course webpage at the same time. These are the lectures for the week.
  - You are expected to check the webpage for the links and listen to the lectures for the week on your own at the scheduled class hours.
    - This will be an offline activity for you with no supervision by us. We will not stream the recordings, you will stream it from youtube directly. We understand that it gives you the chance to listen to the lectures at some other times (or maybe not listen to them at all ☺). However, it is strongly recommended that you use the class timings for this so that you make regular progress. Please understand that learning is a two-way process, and skipping lectures only hurts you.

- The slides for the lectures, if any, will be put up on the course webpage by Wednesday night.
- You can then read up on the topics covered on your own from the text. In case of any doubts, questions etc., you are expected to put them in a google form (link will be sent by mail) by Saturday of that week.
- In the subsequent Monday's class (i.e. next week's Monday), we will answer your questions, clear your doubts, and if necessary go through some topics again from the previous week's coverage. We will also use this time to do tutorial problems and other activities as needed.
  - This session will be conducted online by both instructors. Meeting links will be sent by the instructors.
- TAs will also be available to answer your queries as you study. Each TA will be assigned to a bunch of students (the allocation will be given on the course page soon).
- And of course, you can always directly contact us, your instructors, in case of any help needed or for any other issue you want to discuss with us…..

Evaluation Scheme

- This is TENTATIVE and may change somewhat based on institute guidelines to come…but gives you an idea
- About 60% of your marks will come from 3 to 4 class tests (about 1 hour duration each) conducted online. These test will be announced 1-2 weeks in advance. All submissions will be through Moodle.
- About 40% marks will come from other assessments such as short duration online tests conducted in Moodle, assignments, programming assignments etc.
  - The short duration online tests in Moodle will be held during the discussion hours and will be announced only just before or at the start of the session.

Your Responsibilities

- READ THIS CAREFULLY, please do not give excuses later…
  - If you do not have an account in CSE dept.'s Moodle server (follow the link for Moodle from cse.iitkgp.ac.in), create an account now (see instructions for creating new accounts in CSE Moodle page).
  - After you have an account in CSE dept.'s Moodle server, join the Algorithms-II course for this semester using the enrollment key sent to you in mail.
  - In any case, you must be enrolled in the Algorithms-II course in CSE's Moodle server within the first week of class. If you have any problems, it is your responsibility to contact us within the first week of class and resolve the problem.
  - Irrespective of whether you already have an account or you create a new account, you MUST fill up the "ID Number" field in your Moodle account profile with your roll no. (Go to Edit Profile -> Optional and look for the field) if not already filled.

- The online discussion sessions on Monday will be conducted through Zoom or Microsoft Team. Please download and install the zoom client on your laptop/desktop. For Microsoft Team, you can join through browser. We will inform you about the mode and send the link before the session.
- Attendance is mandatory in the discussion sessions on Monday
  - If there is a test scheduled during the discussion session and you are not present, there will be no compensatory test later
  - We understand you can easily fool us for the attendance ☺, but in your own interest, you should try to attend these sessions….
- For fixed-time online tests, all submissions will be through Moodle. These can be of two types:
  - Tests taken directly on Moodle
  - Tests in which we will ask you to write answers on paper (like normal written tests) or type in the computer, and upload a single text file or a scanned pdf file in Moodle (required file format will be specified).

- For the online tests, we will set the times keeping in consideration different factors like scanning time, slow network speed while submission etc. etc. However, please note that you MUST submit within the cutoff time announced for the test. So if you continue writing till the last minute and do not find time to scan or submit within the time, it is your problem. Think of the last 5 minutes of the test as for all these activities, not for writing answers.
  - In some extreme cases, if you fail to submit within time in Moodle, we may accept an email submission but it will definitely incur a heavy penalty (at least 25%, most likely more). Note that allowing submissions by email, even in extreme cases, is at our discretion and may not be allowed at all, even with penalty. So you are strongly advised to keep the time in mind while taking the tests.
- Copying and other malpractice, if detected, will be dealt with very very strongly. Please note that copying and helping others to copy means the same to us and will incur heavy penalties on BOTH parties, no excuses.

- Please follow the course webpage regularly. While we will usually communicate with you also through email and Moodle, the course webpage is the definitive source for course information, we will post any new notice etc. there.

Course Background

- You have all done Algorithms-I. Major topics that have been covered there include
  - Asymptotic complexity analysis
  - Basic data structures like BST, Heap, Hash Tables, Graphs etc.
  - Advanced data structures: Balanced BST, B-Tree, Disjoint Set Data Structure (Union-Find)
  - Searching and Sorting
  - Algorithm Design Techniques: Divide-and-Conquer, Dynamic Programming, Greedy
  - Graph Algorithms: Representation of graphs, DFS, BFS, MST, Shortest Path (Single-source and All-pairs)
  - String matching algorithms

What we will do in this course

- Amortized complexity analysis with examples
  - Different from the type of analysis you used so far
  - We will look at different analysis techniques with small examples, and then look at an amortized data structure called Fibonacci Heap
- Explore more graph algorithms
  - Graphs are one of the most important tools for modeling practical applications
  - We will look at some problems in two important areas
    - Network Flows – models flow of "item" through a network
    - Bipartite Matching – models many assignment problems where one set of things have to be assigned to another set

- Explore a new domain, Geometric algorithms
  - Algorithms for geometry related problems
  - Will look at algorithms for finding important geometric structures such as convex hulls and voronoi diagrams
- Study the notion of "hardness" of a problem
  - Not all problems have known deterministic polynomial time algorithms
  - We will look at formal notions of specifying hardness of a problem, classification of problems based on their hardness, and how to prove a problem is hard
    - The classes P and NP, NP-hardness/NP-completeness, polynomial reductions
  - Important in practice as if you can prove a problem is NP-hard/NP-complete, the chance of any deterministic polynomial time algorithm existing for it even in future is very very low
    - So you should not try to design one

- Study algorithm design techniques to address these hard problems
  - Ideally we want a deterministic algorithm giving correct answers always in worst case polynomial time
  - But a NP-hard problem does not have one, and chance of it existing in future is very very low
  - But many many important real world problems, especially many optimization problems are NP-hard
    - How do we try to solve them?
  - Approach:
    - Compromise on one of the three: "deterministic, "always correct", "worst case polynomial time"

- We will study new algorithm design techniques that work on the above approaches
  - Branch-and-Bound/Backtracking
  - Randomized Algorithms
  - Approximation Algorithms

- We will start with the first topic, Amortized Analysis, in the next lecture