Boston University Metropolitan College

Analysis of Algorithms A1

CS 566

On Campus

Chertushkin Mikhail
mikhaild@bu.edu
Office hours: by appointment

Course Description

This course covers Analysis of Algorithms in a practically oriented way using Python as the primary programming language. Students will implement algorithms in class and at home. Students will receive practical knowledge that will allow them to successfully design and analyze modern computer science algorithms. This course establishes the relationship between algorithms and programming and teaches the concepts of algorithmic analysis.

Books

Cormen, T. H. & Leiserson C. E. (2009). Introduction to Algorithms, third edition. The MIT Press. Retrieved from: https://www.amazon.com/Introduction-Algorithms-3rd-MIT-Press/dp/0262033844/

Courseware

Blackboard link
Leetcode link
Codeforces link

Class Policies

1) **Attendance & Absences** – this course emphasizes a lot on practice and requires full attendance on lectures. Working laptops with full charge batteries are necessary as they are needed for passing the in-lecture submissions. During all lectures students are going to implement at least one of the algorithms that is covered by theoretical material. The lectures will consist of 50% theory and 50% practice and will be organized as "Reverse Seminars" - this means that students first are presented with an algorithmic problem and then they try to solve it. After trial-and-error students get familiar with necessary theoretical concepts and submit the solution to the grading system after the lecture.
2) **Assignment Completion & Late Work** – every week students will have to pass algorithmic problems in-lecture (easy/medium/hard level problems) and algorithmic problems in homework (medium level problem or theoretical analysis of given algorithm). The time for submission of algorithmic in-lecture problem is 3 days, by Friday the same week 11:59 PM. The time for submission of homework algorithmic

problem or theoretical analysis is 6 days, by Tuesday the next week 11:59 PM. Late submissions are not possible.

3) **Academic Conduct Code** – Cheating and plagiarism will not be tolerated in any Metropolitan College course. They will result in no credit for the assignment or examination and may lead to disciplinary actions. Please take the time to review the Student Academic Conduct Code:

http://www.bu.edu/met/metropolitan_college_people/student/resources/conduct/code.html. This should not be understood as a discouragement for discussing the material or your particular approach to a problem with other students in the class. On the contrary – you should share your thoughts, questions and solutions. Naturally, if you choose to work in a group, you will be expected to come up with more than one and highly original solutions rather than the same mistakes."

Grading Criteria

Grades are calculated as a weighted combination of all in-class problems, homework algorithmic problems and theoretical problems. In-class problems contribute to 50% of grade and homework problems together with theoretical problems contribute to another 50% of grade.

Class Meetings, Lectures & Assignments

There will be lectures every week for the following set of topics. We will examine various topics starting from basics such as growth of functions and elementary data structures to advanced data structures and NP-completeness.

Lectures, Readings, and Assignments subject to change, and will be announced in class as applicable within a reasonable time frame.

Date	Topic	Readings Due	Assignments Due
September 6	Introduction to Algorithms. Two approaches to teaching. Growth of Functions. Elementary Data Structures. Linked Lists and Arrays.	Chapters. 1, 2, 3	n/a
September 13	Sorting. Insertion Sort, Selection Sort and Bubble Sort. Recursion, Tail Recursion and Divide-And-Conquer. Master Theorem.	Ch. 4	Assignments of first week
September 20	Merge Sort. Quicksort. Heapsort. Bucket Sort. Master Theorem, continued	Ch. 6, 7	Assignments of second week

September 27	Linear Sorting. Counting Sort. Bucket Sort. Medians and Order statistics	Ch 8, 9	Assignments of third week
October 4	Hast Tables. Hash Maps. Collision problems.	Ch 11	Assignments of fourth week
October 11	Binary Search. 3 use-cases of Binary Search.	N/A	Assignments of fifth week
October 18	Binary Trees. Binary Search Trees. Red-Black trees and AVL trees.	Ch 12, 13	Assignments of sixth week
October 25	Dynamic programming. Top-down approach and bottom-up approach.	Ch 15	Assignments of seventh week
November 1	Greedy Algorithms. Global optimality problem.	Ch 16	Assignments of eight week
November 8	Graph Algorithms. Minimum Spanning Trees.	Ch 22, 23	Assignments of ninth week
November 15	Matrix Operations	Ch 28	Assignments of tenth week
November 22	Thanksgiving break	Thanksgiving break	Thanksgiving break
November 29	String Matching. Aho-Corasick algorithm. Knutt-Moris-Pratt algorithm	Ch 32	Assignments of eleventh week
December 6	NP-completeness	Ch 34	n/a