

University American College Skopje
School of Computer Science and Information Technology

Academic 2024/25

Version: 25.09.2024

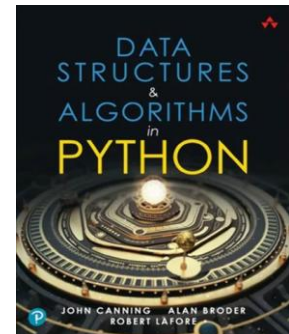
Course: Data Structures and Algorithms
Course Code: CSCI 2331
Semester: Third
Year: II
Prerequisites: Object-oriented Programming
Lecturer: Igor Lazov, PhD (igor.lazov@uacs.edu.mk)
Collaborator: Marija Stankova Medarovska, PhD
(marija.s.medarovska@uacs.edu.mk)

Duration of Course: Theory 30 hours + Practical 30 hours
Credit Hours: 6 ECTS
Schedule hours:
Room:
Meeting hours:
Consultation hours: Lectures: Wednesday 11:00 - 12:00
Exercises: Thursday 13:30 - 14:30

Required text: Books and Other Learning Materials Available on the Moodle Course

Data Structures and Algorithms in Python (Developer's Library) 1st Edition

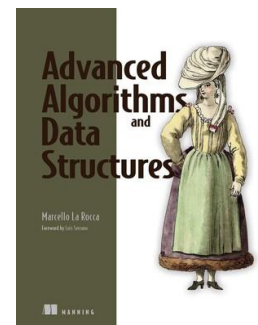
by John Canning, Alan Broder, Robert Lafore
Released: October 2022
Publisher(s): Addison-Wesley Professional
ISBN: 978-0-13-485568-4



Advanced Algorithms and Data Structures

by Marcello La Rocca

Released: June 2021
Publisher: Manning Publications Co.
ISBN: 9781617295485



Supplementary reading: Extra Learning Materials Available on the Moodle Course

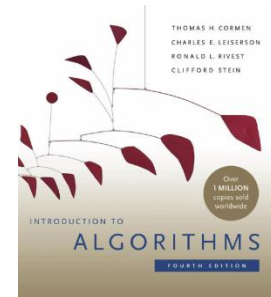
Introduction to Algorithms

*by Thomas H. Cormen, Charles E. Leiserson,
Ronald L. Rivest, and Clifford Stein*

Published: April 5, 2022

Publisher: The MIT Press

ISBN: 9780262046305



Course outline / Description: The Course will give a complete overview of algorithms and data structures from a practical standpoint. The course begins with explaining data files and working with them. Then, algorithms are introduced, as well as algorithmic complexity, followed by showing the most standard algorithms for sorting, both iterative and recursive. Then functional and class templates are introduced, which are used to explain working with linked lists, as well as algorithms for their sorting. In the second part, stacks, queues and trees are introduced. Also, standard algorithms for working with these structures are given. Finally, special attention is given to graphs, as especially important structures, as well as algorithms used upon working with graphs. The algorithms will be implemented in the programming language C++, so a prior C++ programming knowledge is needed.

Learning objectives: The goal of this course is to help students develop algorithmic thinking. Thereby, they will learn how to analyze the various Sorting and Searching algorithms and techniques. Then, how to implement the abstract data types for Linear data structures (Lists, Stacks, Queues) and Non-linear data structures (Trees, Graphs), and also apply them to problem solutions. At the end of the course, the student should be able to:

- work with files;
- get acquainted and work with basic sorting techniques;
- understand function and class templates;
- manage working with linked lists;
- know what are stacks and queues and to be able to work with them;
- understand the notion of recursion and be able to implement it;
- know what are binary trees and to be able to work with them;
- understand graphs, differentiate among various types of graphs and be able to work with them.

Course delivery and format: The course will be delivered through lectures and exercises. The materials will be presented through ppt presentations and pdf tutorial-like files. After each lecture and/or exercise there will be a participation quiz for theory questions from the lectures and/or practical questions from the exercises in view of the corresponding material. The exercises will include practice algorithms, C++ code, and analysis of abstract data types. Furthermore, for the exercises the practical implementations will be demonstrated in adequate software that will be available in the computer laboratories. The course will introduce homework assignments that are presented as mandatory for home practice and are not for evaluation, however, small positive competition between students will be presented for extra / bonus points concerning the homework each week that will influence the final evaluation of the student. After mid-term, the students will need to start preparing for a course project divided in groups which they need to complete and deliver by the end of the 13th week of the course after which on the 14th week they need to verbally present their work before the class for evaluation, otherwise it won't be considered for evaluation.

Learning outcomes:

1. Knowledge and Understanding:

By the end of this course, students will have a solid foundation in data structures and algorithms, understanding key concepts of basic programming constructs, like loops, conditionals, and functions. They will learn to translate real-world problems into structured steps and apply these steps to create functional algorithms. Additionally, students will develop a deep understanding of fundamental programming concepts, such as data types, control structures, and the basic operations that form the backbone of programming languages, like Python. Also, they will implement the data structures, like linked lists, stacks, queues, trees, and graphs.

2. Applying Knowledge and Understanding:

Students will apply the knowledge gained in class through hands-on exercises and practical assignments. They will practice developing algorithms into Python programs. By engaging with real-world scenarios, students will enhance their problem-solving skills, implementing algorithms to tackle searching, sorting, and other tasks. Practical applications will be demonstrated through software tools, such as Visual Studio Code, further refining their programming capabilities.

3. Making Judgements:

Throughout the course, students will be encouraged to critically evaluate their approach to solving programming tasks. They will make informed decisions about the most efficient algorithmic methods for given problems, considering factors such as: time and space complexity, optimization, and functionality. By analysing different strategies, they will be able to assess their solutions' accuracy, efficiency, and scalability, developing their ability to choose the best algorithmic and coding practices for various contexts.

4. Communication Skills:

Students will enhance their communication skills through both written and verbal formats. They will present their understanding of data structures, algorithms, and programming concepts in a clear, structured manner. Group projects will offer opportunities for collaboration, where students will articulate their ideas and findings. Presenting their project outcomes at the end of the course will allow them to refine their ability to explain technical concepts in an accessible way, promoting effective communication within team environments.

5. Learning Skills:

The course will foster independent learning skills, as students will engage in self-guided practice through homework and project-based tasks. By navigating software tools, solving problems, and exploring various algorithms, students will build a robust framework for lifelong learning in programming. The inclusion of quizzes will motivate students to take initiative in their learning, enhancing their ability to adapt and acquire new skills throughout their programming journey.

Instructor's expectations from students / Students' responsibilities:

- Regular attendance of lectures and exercises on-site.
- Timely arrival at classes.
- Careful follow-up of lectures and exercises.
- Academic behaviour and preservation of the working atmosphere of classes.
- Mandatory remarks about everything that is written and spoken in classes.
- Active participation in the discussion of the lessons by completing the provided quizzes.
- Timely preparation and mandatory oral presentation of semestral projects.
- Independence and honesty in work.
- Friendly competition between students for acquiring extra bonus points from homeworks presented after each exercise.
- Behaviour with respect to the Code of Ethics of the University.
- If online sessions are scheduled on special occasions, the students are required to turn On their camera at all times during the class.

Weekly delivery plan:

Class	Chapter	Lecture Topics	Exercises Topics (Self Assessment Questions – Quizzes, Discussion Questions, Cases) from the Textbook
1		Data structures	<u>Introduction to Python.</u>
2		Algorithms and their classifications	<u>Data types.</u>
3		Algorithm analysis, O-notation	<u>Functions.</u> Parameters, recursion.
4		Searching techniques	<u>Searching.</u> Linear search, binary search.
5		Sorting techniques	<u>Sorting.</u> Selection sort, insertion sort, bubble sort, quick sort, merge sort.
6	Trial Mid-Term Exam		

7	Mid-Term Exam		
8		Link list, Doubly link list, Circular link list	<u>Linked lists</u> . Algorithms with linked lists.
9		Stacks and Queues	<u>Abstract data types</u> . Stacks. Queues.
10		Tree data structure	<u>Trees</u> . Binary Trees. Binary Search Trees
11		Tree traversal, AVL tree	<u>Tree traversal</u> . Tree Rotation. Tree Roll
12		Graph data structure	<u>Graphs</u> . Graph search. Directed graphs. Directed acyclic graphs
13		Spanning trees	<u>Minimum spanning trees</u> .
14	Project Presentations		
15	Reading Week		
16	Final-Term Exam		

Evaluation structure and assessments during teaching semester:

- Attendance criteria:
 - Up to 25% absences (4 of 15) – can get points and be evaluated;
 - 25-50% absences (5 - 7 of 15) - no points from attendance (0% attendance evaluation) but will be allowed to take exams;
 - More than 50% absence (8 and above) - is not allowed to take the Final-Term Exam, in a sense, it will not be possible to come to the Final-Term Exam and to be graded if minimum criteria are not met.
 - In this case, the student must come directly to the MakeUp Exam at the end of the semester.
- Exams criteria:
 - Minimum requirements of 20% of the Mid-Term Exam (20 of 100 points) are necessary for the student to be eligible in taking the Final-Term Exam.
 - Minimum requirements of 20% of the Final-Term Exam (20 of 100 points) are necessary for the student to be eligible for final Evaluation and making of the final grade.
 - In a sense, it will not be possible to take the Final-Term Exam and to be graded if minimum criteria from both attendance and exam are not met.
 - In this case, the student must come directly to the MakeUp Exam at the end of the semester.
- Regular participation and activity with attendance: 20%, of which:
 - Attendance on classes: 10%
 - Active participation on quizzes: 10%
- Mid-Term Exam: 30% of 100 points gathered below, of which it will include:
 - Theory Quiz from the Lectures done via SEB (Safe Exam Browser)
 - 40 points of 100
 - Practical Questions Quiz from the Exercises done via SEB
 - 20 points of 100
 - Practical Assignments (one small simple assignment on pencil and paper, and the second more complex assignment on a computer device in a corresponding software from the exercises).
 - 40 points of 100
 - Important Note: To avoid cheating and misuse of various AI tools, the second assignment will not be evaluated if the first assignment is not completed or not correct

as it is the initial evaluation of the student to recognize that he/she understands the materials.

- Exams quizzes, both for theory and practical, consist of questions similar to the questions from the active participation quizzes done after each lecture and exercise.
 - The Exam Practical Assignments consist of similar practice assignments done on each of the exercises and from the homework practice assignments.
 - Total:
 - Theory Quiz (40 points) + Practical Questions Quiz (20 points) + Practical Assignments (40 points) = 100 points
- Final-Term Exam: 30% of 100 points gathered below, of which it will include:
 - Theory Quiz from the Lectures done via SEB (Safe Exam Browser)
 - 40 points of 100
 - Practical Questions Quiz from the Exercises done via SEB
 - 20 points of 100
 - Practical Assignments (one small simple assignment on pencil and paper, and the second more complex assignment on a computer device in a corresponding software from the exercises).
 - 40 points of 100
 - Important Note: To avoid cheating and misuse of various AI tools, the second assignment will not be evaluated if the first assignment is not completed or not correct as it is the initial evaluation of the student to recognize that he/she understands the materials.
 - Exams quizzes, both for theory and practical, consist of questions similar to the questions from the active participation quizzes done after each lecture and exercise.
 - The Exam Practical Assignments consist of similar practice assignments done on each of the exercises and from the homework practice assignments.
 - Total:
 - Theory Quiz (40 points) + Practical Questions Quiz (20 points) + Practical Assignments (40 points) = 100 points
 - Semester Course Project: - 20% of 100 points gathered below, of which it will include:
 - With group preparation of the seminar tasks (students will be divided in groups according to the professor), students will prepare their project assignment by submitting a documentation of the solved project assignment and PowerPoint presentation in which all of the group members need to orally present their participation and work in the project.
 - Project Submission: 10% (50 points)
 - Oral Presentation: 10% (50 points)
 - Important Note: Oral presentation of the project is mandatory in order to be included for evaluation.
 - Total:
 - Project Submission (10% - 50 points) + Oral Presentation (10% - 50 points) = 20% (100 points)
 - Assessment structure for the course:

○ <i>Assignments</i>	<i>% of total Grade</i>
○ Class Presence	10%
○ Active Participation (Quizzes)	10%
○ Project or assignment	20%
○ Midterm Exam	30%
○ Final Exam	30%
○	
○ <u>Total:</u>	<u>100%</u>

Evaluation structure and assessments for MakeUp examinations

- **MakeUp Exam: 100%**, of which it will include:
 - Theory Quiz from the Lectures done via SEB (Safe Exam Browser)
 - 40%
 - Practical Questions Quiz from the Exercises done via SEB
 - 20%
 - Practical Assignments (one small simple assignment on pencil and paper, and the second more complex assignment on a computer device in a corresponding software from the exercises).
 - 40%
 - **Important Note:** To avoid cheating and misuse of various AI tools, the second assignment will not be evaluated if the first assignment is not completed or not correct as it is the initial evaluation of the student to recognize that he/she understands the materials.
 - Exams quizzes, both for theory and practical, consist of questions similar to the questions from the active participation quizzes done after each lecture and exercise.
 - The Exam Practical Assignments consist of similar practice assignments done on each of the exercises and from the homework practice assignments.
- **Total:**
 - Theory Quiz (40%) + Practical Questions Quiz (20%) + Practical Assignments (40%) = 100%

Evaluation structure and assessments for Part-Time Student examinations:

- **Exam: 100%**, of which it will include:
 - Theory Quiz from the Lectures done via SEB (Safe Exam Browser)
 - 40%
 - Practical Questions Quiz from the Exercises done via SEB
 - 20%
 - Practical Assignments (one small simple assignment on pencil and paper, and the second more complex assignment on a computer device in a corresponding software from the exercises).
 - 40%
 - **Important Note:** To avoid cheating and misuse of various AI tools, the second assignment will not be evaluated if the first assignment is not completed or not correct as it is the initial evaluation of the student to recognize that he/she understands the materials.
 - Exams quizzes, both for theory and practical, consist of questions similar to the questions from the active participation quizzes done after each lecture and exercise.
 - The Exam Practical Assignments consist of similar practice assignments done on each of the exercises and from the homework practice assignments.
- **Total:**
 - Theory Quiz (40%) + Practical Questions Quiz (20%) + Practical Assignments (40%) = 100%

Exams are closed books. Also, you use your own calculator, pencil, and empty paper signed by the professor, and nothing else will be allowed.

Mobile phones are strictly not tolerated in the class for any use (including computations). Active participation is meant as the effort and the interest that a student shows in the class, including homework.

Make-up tests are given in exceptional circumstances. Cheating and plagiarism in any form will result immediately in the grade F. I assign a grade of (incomplete) only when a student misses the Midterm or the Final exam due to a **College-excused absence**, and is unable to make up before final

grades are due to the Record's Office. Please, consider seriously that there will be no exceptions concerning the above policy.

Class Conduct : You are responsible for everything that is announced, presented or discussed in class. The way to avoid any misunderstanding associated with this course is to attend class. You are expected to attend class and attendance records are kept.

Please, refrain from talking during class, it is disruptive to your colleagues and the lecture. Give only your best behaviour in class. This is what education is all about. If you have a question about the material, please don't hesitate to ask; if you are lost or confused, your question may help clarify the topic. Please, consider that **the language of instruction is English, so all our conversation into the class must be in this language.**

Keep in mind the following :

C- or better is required to use a course either as a prerequisite or as a major requirement.

G.P.A. (Grade Point Average): is computed for each student using the quality points earned for each course taken. A G.P.A. of at least 1.67 is required for transfer and to graduate from ACS.

I wish you an interesting and creative academic semester.

Academic Honesty

Academic honesty is a core value that safeguards the reputation of University American College Skopje and its faculty, staff, and students.

All UACS members are expected to conduct themselves in an honest manner observing the University Code of Ethics.

Plagiarism, cheating, multiple submissions, falsification of data, and unauthorised assistance are considered as serious offences and may result in grade F (fail) on the exam, project, and course, and/or lead to further sanctions, including expulsion from the University.

Any work that is submitted at any stage must be your own. Any words, ideas, or data borrowed from other sources must be properly credited and documented.

You may not use generative AI tools, such as ChatGPT, while completing any assignment, unless explicitly permitted by the course instructor in specific contexts. However, you may use generative AI for self-learning purposes on assignments and exercises only if instructed by the course instructor for exercises and home learning. Any use of generative AI tools outside of these parameters constitutes plagiarism and will be sanctioned accordingly.

Late Work

Many reasons are given for late work (e.g., flat tires, printer problems, illness, oversleeping, etc.). No matter what the reason, the bottom line is that either (a) the work was completed and turned in when required or (b) the assignment was not completed and turned in when required.

As specified earlier, assignments are due in class at the beginning of class on the specified due date. There is NO PROVISION for late work on any assignment (i.e., late work is not accepted). I also strongly recommend having backup systems in place so that you can have all work completed on schedule. Having your work completed on schedule is a key to early success in your business career. NOTE: Late submissions are not accepted. Partial credit will NOT be given for late work.

Prepared by:

Igor Lazov, PhD

Marija Stankova Medarovska, PhD

Evaluation		
Mark	%	
Average		
A	96-100	4.00
A-	90-95	3.67
B+	87-89	3.33
B	83-86	3.00
B-	80-82	2.67
C+	77-79	2.33
C	73-76	2.00
C-	70-72	1.67
D+	67-69	1.33
D	63-66	1.00
D-	60-62	0.67