

ALGORITHMS AND DATA STRUCTURES

COURSE SYLLABUS

Code and name of specialty	122 — Computer Science	Institute / faculty	Faculty of Computer Science and Software Engineering
Program name	«Computer Science and Intelligent Systems»	Department	Software Engineering and Management Information Technologies
Type of program	Educational and Professional	Language of instruction	Ukrainian, English

LECTURER

Name, E-mail	Nataliia Stratiienko, Nataliia.Stratiienko@kphi.edu.ua
---------------------	---



Ph.D., Associate Professor, Professor of the Department of Software Engineering and Management Information Technologies, NTU «KhPI». Prepared and published more than 60 research papers, training manuals and textbooks (1 training manual recommended by the Ministry of Education and Science of Ukraine, 1 training manual recommended by the Academic Council of NTU "KhPI", 3 articles in publications indexed in Scopus) (Google Scholar: <https://scholar.google.com/citations?user=9cw0zwwgAAAAJ&hl=ru>; ORCID: <https://orcid.org/0000-0002-7925-6687>; Scopus: <https://www.scopus.com/authid/detail.uri?authorId=57196007565>).

Leading lecturer of courses: *Fundamentals of the Theory of Algorithms, Algorithms and Data Structures, Computer Mathematics, Fundamentals of Project Management, Formation and Development of IT Project Teams (in Ukrainian and English)*

GENERAL DESCRIPTION OF THE COURSE

Summary	The course “Algorithms and Data Structures” is a course in the cycle of professional compulsory training of the specialty 122 “Computer Science”. It is taught in the second semester in the amount of 120 hours (4 ECTS credits), in particular: lectures – 32 hours, laboratory classes – 32 hours, self-study work – 56 hours. The course includes two modules and two modular tests. The study of the discipline ends with the test.
Course objectives	This course objective is formation of students' knowledge of basic data structures and algorithms as well as the acquisition of practical skills in the analysis of algorithms.
Types of classes and control	Lectures, laboratory classes, self-study work (and individual calculation work). Final assessment – test.
Term	2

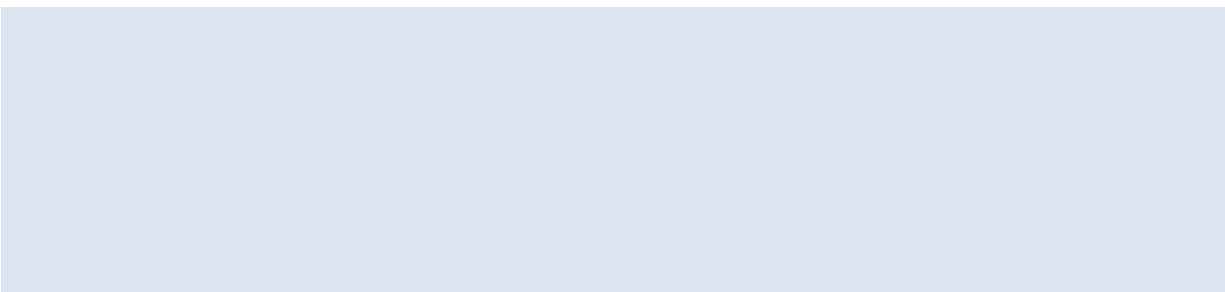
Student workload (credits) / Type of course	4 / Mandatory	Lectures (hours)	32	Laboratory classes (hours)	32	Self-study (hours)	56
--	---------------	-------------------------	----	-----------------------------------	----	---------------------------	----

Program competences	<p>GC1. Ability to abstract thinking, analysis and synthesis.</p> <p>GC2. Ability to apply knowledge in practical situations.</p> <p>GC3. Knowledge and understanding of the subject area and understanding of professional activity.</p> <p>GC6. Ability to learn and master modern knowledge.</p> <p>GC7. Ability to search, process and analyze information from various sources.</p> <p>PC3. Ability to think logically, build logical conclusions, use formal languages and models of algorithmic calculations, design, develop and analyze algorithms, evaluate their efficiency and complexity, solvability and unsolvability of algorithmic problems for adequate modelling of subject areas and creation of software and information systems.</p> <p>PC8. Ability to design and develop software using different programming paradigms: generalized, object-oriented, functional, logical, with appropriate models, methods and algorithms of calculations, data structures and management mechanisms.</p>
----------------------------	---

Learning outcomes	Teaching and learning methods	Forms of assessment (continuous assessment CAS, final assessment FAS)
PLO1. Apply knowledge of the fundamental forms and laws of abstract-logical thinking, the basics of the methodology of scientific knowledge, forms and methods of extraction, analysis, processing, and synthesis of information in the subject area of computer science.	Interactive lectures with presentations, discussions, laboratory classes, teamwork, case method, feedback method from students, problem learning	Written individual assignments for laboratory work (CAS), assessment of knowledge in laboratory classes (CAS), express - survey(CAS), online tests (CAS), final / semester control in the form of a semester test, in accordance with the schedule of the educational process (FAS)
PLO5. Design, develop and analyze algorithms for solving computational and logical problems, evaluate the efficiency and complexity of algorithms based on the use of formal models of algorithms and computational functions.	Interactive lectures with presentations, discussions, laboratory classes, teamwork, case method, feedback method from students, problem learning	Written individual assignments for laboratory work (CAS), assessment of knowledge in laboratory classes (CAS), express - survey(CAS), online tests (CAS), final / semester control in the form of a semester test, in accordance with the schedule of the educational process (FAS)
PLO9. Develop software models of subject areas, choose a programming	Interactive lectures with presentations, discussions, laboratory classes, teamwork, case method,	Written individual assignments for laboratory work (CAS), assessment of knowledge in laboratory classes (CAS), express - survey(CAS), online tests (CAS), final / semester control in the form of a semester test, in accordance with the schedule of the educational process (FAS)

paradigm from the standpoint of convenience and quality of its application to implement methods and algorithms that solve problems in the computer science field.

feedback method from students, problem learning



ASSESSMENT AND GRADING

Ranges of points corresponding to grades	Total score (points) for all types of learning activities	ECTS grading scale	The national grading scale	Allocation of grade points	100% Final assessment as a result of Final test (30%) and Continuous assessment (70%). 30% Final test 70% Continuous assessment: Laboratory classes (18%) Calculation work (22%) Two module tests (30%)
	90-100	A	excellent		
	82-89	B	good		
	74-81	C			
	64-73	D	satisfactory		
	60-63	E			
	35-59	FX	Unsatisfactory (with the exam retake option)		
	0-34	F	Unsatisfactory (with mandatory repetition of the course)		

Course policy Students are required to attend classes as scheduled and comply with ethical conduct. If absent, students will need to complete all tasks to compensate for the missed classes. Participation in laboratory work requires preliminary preparation and advance processing of all the necessary materials for productive work during the lesson. Written assignments must be submitted on time.

COURSE STRUCTURE AND CONTENT

Topic 1	Introduction to data structures and algorithms	Laboratory work 1	Basic data structures (list, queue, stack)	Самостійна робота	.Individual calculation work.
Topic 2	Basic data structures	Laboratory work 2	Basic data structures. Hash tables		
Topic 3	Sorting, merging, and search	Laboratory work 3	Basic data structures. Red-black trees		Studying the course topics with the help of recommended reading, homework
Topic 4	Combinatorial algorithms	Laboratory work 4	Sorting algorithms		
Topic 5	Fundamental graph	Laboratory work 5	Combinatorial algorithms		

	algorithms				
Topic 6	Geometrical algorithms	Laboratory work 6	Fundamental algorithms on graphs and trees		
Topic 7	Cryptographic algorithms	Laboratory work 7	Geometrical algorithms		
Topic 8	Approximation algorithms	Laboratory work 8	Dynamic programming		
Topic 9	Mathematical foundations of algorithm analysis	Laboratory work 9	Greedy algorithms		
Topic 10	Recursion				
Topic 11	Algorithmic strategies				
Topic 12	Basics of computability theory				
Topic 13	Classes P and NP				

RECOMMENDED READING

1. Marcello La Rocca. (2021) Advanced Algorithms and Data Structures. New York: Manning Publications Co.
2. Крєневич А.П. (2021) Алгоритми і структури даних. Підручник. Київ: ВПЦ "Київський Університет".
3. Helmut Knebl. Algorithms and Data Structures: Foundations and Probabilistic Methods for Design and Analysis (2020) Cham: Springer Nature Switzerland AG
4. Стратієнко, Н.К., Годлевський, М.Д., Бородіна І.О. (2017) Алгоритми і структури даних: практикум:Харьков: НТУ"ХПИ"
5. Стратієнко, Н. К., Бородіна І. О. (2017) Методичні вказівки до виконання лабораторних робіт з курсу "Алгоритми і структури даних" : для студ., які навч. за спец. 121 "Інженерія програмного забезпечення" Харківський політехнічний ін-т, нац. техн. ун-т. Електрон. текстові дані. Харків,

6. Donald Knuth. (2020) The Art of Computer Programming, Volume 4, Fascicle 5: Mathematical Preliminaries Redux; Introduction to Backtracking. Boston: Pearson Education (US).
7. Florian Jatton, Geoffrey C. Bowker. (2021) The Constitution of Algorithms: Ground-Truthing, Programming, Formulating. MIT Press Ltd, United States.
8. Shmuel Tomi Klein. (2021) Basic Concepts In Algorithms. / Shmuel Tomi Klein. – Singapore: World Scientific Publishing Co Pte Ltd.
9. Hemant Jain.(2019) Problem Solving in Data Structures & Algorithms Using Python. /Hemant Jain. – Independently Published.
10. Hemant Jain (2018) Problem Solving in Data Structures & Algorithms Using C. Independently Published.
11. Steven S. (2020) Skiena. The Algorithm Design Manual. 3rd ed. Cham: Springer Nature Switzerland AG.
12. Мелешко Є.В., Якименко М.С., Поліщук Л.І. (2019) Алгоритми та структури даних: Навчальний посібник для студентів технічних спеціальностей денної та заочної форми навчання. Кропивницький: Видавець – Лисенко В.Ф.
13. Ільман В.М., Іванов, О.П., Панік. Л.О. (2019) Алгоритми, дані і структури. Дніпропет. нац. ун-т залізн. трансп.ім. акад. В. Лазаряна. Дніпро
14. Прийма С.М. (2018) Теорія алгоритмів: Навчальний посібник. Мелітополь: ФОП Однорог Т.В.
15. Бородкіна І.Л. (2018) Теорія алгоритмів: посібник для студентів вищих навчальних закладів. Центр навчальної літератури (ЦУЛ).
16. Allen Downey., (2017) Think Data Structures O'Reilly Media, Inc, USA.
17. Marcin Jamro. (2018) C# Data Structures and Algorithms: Explore the possibilities of C# for developing a variety of efficient applications Birmingham: Packt Publishing Limited.
18. Стратієнко, Н. К., Шматко О. В., Бородіна О. (2016) Методичні вказівки до виконання курсової роботи по курсу "Алгоритми та структури даних" : для студ., які навч. за напрямком 6.050103 "Програмна інженерія" спец. 05010301 "Програмне забезпечення систем".; Харківський політехнічний ін-т, нац. техн. ун-т. Електрон. текстові дані. Харків. Retrieved from: <http://repository.kpi.kharkov.ua/handle/KhPI-Press/24697>.
- 19 Stratiienko N. K. Borodina I. O. (2016) Guidance for course work on "Algorithms and Data Structures" : for students of direction 6.050103 "Software Engineering", specialty .05010302 "Software Engineering" = Методичні вказівки до виконання курсової роботи по курсу "Алгоритми та структури даних" : для студ., які навч. за напрямком 6.050103 "Програмна інженерія" спец. 05010302 "Інженерія програмного забезпечення" [Electronic resource] / comp. ; National Technical University "Kharkiv Polytechnic Institute". Kharkiv, Access mode: Retrieved from: <http://repository.kpi.kharkov.ua/handle/KhPI-Press/24695>.

Academic integrity

Graduate students are expected to adhere to the Code of Ethics of Academic Relations and Integrity” of NTU “KhPI”.

The content of this syllabus is consistent with the course program.