Syllabus

Course Program

Theory of algorithms

Specialty
121 – Software Engineering

Institute
Institute of Computer Science and Information Technology

Educational program
Software Engineering

Department
Software Engineering and Management Intelligent Technologies (321)

Level of education
Bachelor's level

Course type
Special (professional), Mandatory

Semester
3

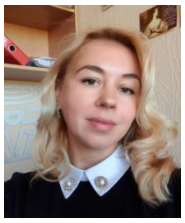
Language of instruction
English, Ukrainian

Lecturers and course developers

Oleksandr Shmatko

oleksandr.shmatko@khpi.edu.ua

Ph.D., Ass. Prof. at the Department of Software Engineering and Management Information Technologies of NTU «KhPI». Prepared and published more than 70 publications, 3 collective monographs, 2 textbooks with the University stamp, 8 articles in publications indexed in Scopus. (h-index = 7, i10-index = 4 in Google Academy-https://scholar.google.com/citations?user=Wyv6esuaaaaaj&hl=ru; ORCID iD-https://orcid.org/0000-0002-2426-900x.
Leading lecturer of courses: Software modeling and analysis (bachelors) (in English and Ukrainian), Advanced technologies and areas of development of intelligent software systems (Masters) (in English and Ukrainian), Modern technologies of web application development (PhD) (in Ukrainian)More about the lecturer on the department's website



Uliya Litvinova

email@khpi.edu.ua

Ph.D. (01.05.02 Mathematical Simulation and Methods of Calculation). Associate Professor at the Department of Software Engineering and Management Information Technology. Work experience – since 2006. Author (co-author) of more than 40 research papers and textbooks. (h-index = 3 in Google Scholar - https://scholar.google.com.ua/citations?user=8cVqocUAAAAJ&hl=uk; ORCID ID is https://orcid.org/0000-0001-6680-662X)
Basic courses: "Fundamentals of Web development" (lectures and lab classes), Innovation and entrepreneurship"(lectures and lab classes), "Scientific and practical workshop Software engineering" "(workshop).
More about the lecturer on the department's website

General information

Summary

Forming a system of knowledge about basic data structures and basic computing algorithms, as well as acquiring practical skills in designing, developing and analysing algorithms, evaluating their efficiency and complexity.

Course objectives and goals

Introduction of students to basic data structures; introduction of students to basic computational algorithms; study of the mathematical apparatus for the study of recursive functions; teaching students the basics of algorithm analysis; study of algorithm complexity classes.

Format of classes

Lectures, laboratory classes, consultations, self-study. Final control in the form of an exam.

Competencies

К01. Ability to think abstractly, analyze and synthesize.
К05. Ability to learn and master modern knowledge.
К06. Ability to search, process and analyze information from various sources.
К20. Ability to apply fundamental and interdisciplinary knowledge to successfully solve software engineering problems.
К26. Ability to think algorithmically and logically..

Learning outcomes

PLO01. Analyze, purposefully search and select information and reference resources and knowledge necessary for solving professional problems, taking into account modern achievements of science and technology.
PLO13. Know and apply methods of developing algorithms, designing software and data structures and knowledge.

Student workload

The total volume of the course is 120 hours (4 ECTS credits): lectures - 28 hours, laboratory classes - 28 hours, self-study - 64 hours.

Course prerequisites

Fundamentals of programming.
Higher mathematics.
Fundamentals of software engineering.

Features of the course, teaching and learning methods, and technologies

interactive lectures with presentations, discussions, laboratory classes, teamwork, case method, student feedback, problem-based learning.
Forms of assessment:
written individual assignments for laboratory work (CAS), assessment of knowledge in laboratory classes (CAS), express surveys (CAS), online tests (CAS), final/semester control in the form of a semester exam, in accordance with the schedule of the educational process (FAS).

Program of the course

Topics of the lectures

Topic 1 Introduction to data structures and algorithms
The concept of data structures and their classification. Formalisation of the concept of algorithm. The main directions in the theory of algorithms. Practical application of the results of the theory of algorithms

Topic 2 Basic data structures

Arrays. Stacks. Queues. Linked lists. Hash tables. Direct addressing. Hash functions. Binary search trees. Red and black trees.

Topic 3 Sorting, merging and searching algorithms

Sorting in quadratic time. Sorting by selection. Sorting by exchange. Sorting in O(n log n) time. Fast sorting. Selecting an element to split. Sorting by merge. Memory consumption during merge sorting. Merge sequences. Binary search. Lower estimates of sorting speed. The decision tree. Sorting by counting.

Topic 4 Combinatorial algorithms

Pseudorandom number generators. Properties of random and pseudorandom numbers. Common disadvantages of pseudorandom number generators. Linear congruent method. Mersenne vortex.

Topic 5 Fundamental algorithms on graphs and trees

Representation of graphs. Vertices. Edges. Oriented and undirected graphs. List of adjacent vertices. Adjacency matrix. Sparse graph. Weighted graph. Search in depth. Search in width. Search trees. Recursive and non-recursive implementations of searching in a graph. Topological sorting

Topic 6 Geometric algorithms

Properties of segments. Convex combination. Vector product. The direction of rotation. Checking the intersection of segments. The bounding rectangle. The ratio of order on segments. Construction of convex hull. Graham's algorithm. Jarvis' algorithm. The method of adding points. Complexity of convex hull construction algorithms.

Topic 7 Cryptographic algorithms

Classification of cryptographic algorithms. Private key cryptosystems. The concept of a private key. Caesar's cipher. The Vigilant cipher. Cracking the Caesar and Vigilant ciphers. Cryptosystems with a public key. The concept of a public key. Euler's function. Cryptographic hash functions. The need for cryptographic encryption.

Topic 8 Heuristic algorithms

Evaluation of the quality of an approximate algorithm. Approximation scheme for a given optimisation problem. The problem of vertex coverage. An approximate algorithm for finding vertex coverage. The maximum error of an approximate algorithm for finding vertex coverage. The traveling salesman's problem. The inequality of a triangle. An approximate algorithm for the traveling salesman problem. Maximum error of an approximate algorithm for the salesman problem.

Topic 9 Mathematical foundations of algorithm analysis

Asymptotic notation. Growth rate of functions. Logarithmic growth. Linear growth. Quadratic growth. Exponential growth. Standard functions and notation. Sums and their properties. Progressions. Sums of differences. Evaluation of sums. Induction. Partial comparison.

Topic 10 Recursion

Algorithmic system based on recursive functions. The method of substitution. Ways to guess the estimate (analogy, successive approximations). Induction and the inventor's paradox. Replacement of variables. Conversion to sums. Iteration of ratios. Recursion tree. General solution of a large class of recurrence relations. Basics.

Topic 11 Algorithmic strategies

Divide and conquer principle. Division into subtasks. Dynamic programming. The problem of matrix multiplication. Estimating the complexity of the algorithm for solving the matrix multiplication problem. The problem of finding the largest common subsequence. The length of the largest common subsequence. Reproduction of the largest common subsequence. Greedy algorithms. The problem of distributing applications. Depreciation analysis. Grouping method. Subscription method. The method of potentials.

Topic 12 Fundamentals of the theory of computability

The concept of computability and computational procedures. The concept of relative algorithm and relative computability, the concept of reduction. Turing machine. Components of the Turing machine. The capabilities of the Turing machine. Illustration of the Turing machine for simple algorithms. The basic hypothesis of the theory of algorithms. Algorithmically intractable problems.

Topic 13 Complexity classes P and NP

Polynomial time. Efficient algorithm. Abstract problem. Polynomial time problem. Formal languages for solvability problems. Language membership checking and NP class. The problem of the Hamiltonian cycle in a graph. Verification algorithm. NP-hard and NP-complete problems. The class of NP

Topics of the workshops

Workshops are not provided within the discipline.

Topics of the laboratory classes

Laboratory work №1. Basic data structures: (list, queue, stack)
Laboratory work №2. Basic data structures: hash tables
Laboratory work №3. Basic data structures: red and black trees
Laboratory work №4. Sorting algorithms
Laboratory work №5. Combinatorial algorithms
Laboratory work №6. Fundamental algorithms on graphs and trees
Laboratory work №7. Geometric algorithms
Laboratory work №8. Dynamic programming
Laboratory work №9. Greedy algorithms

Self-study

The curriculum provides for the completion of a calculation task (CT). At the beginning of the semester, students choose the topics from the list or propose their own topics and agree them with the teacher. The assignment is completed during the semester and defended during the test week or examination session. Students are recommended additional materials (videos, articles) for independent study and processing.

Course materials and recommended reading

Compulsory materials

1. Marcello La Rocca. (2021) Advanced Algorithms and Data Structures. / New York: Manning Publications Co.
2. Helmut Knebl. (2020) Algorithms and Data Structures: Foundations and Probabilistic Methods for Design and Analysis. – Cham: Springer Nature Switzerland AG,

Additional materials

3.Donald Knuth. (2020) The Art of Computer Programming, Volume 4, Fascicle 5: Mathematical Preliminaries Redux; Introduction to Backtracking. Boston: Pearson Education (US),
4. Florian Jaton, Geoffrey C. Bowker. (2021) The Constitution of Algorithms: Ground-Truthing, Programming, Formulating. MIT Press Ltd, United States
5. Shmuel Tomi Klein. (2021) Basic Concepts In Algorithms. Singapore: World Scientific Publishing Co Pte Ltd
6. Hemant Jain.(2019) Problem Solving in Data Structures & Algorithms Using Python. Independently Published,
7. Hemant Jain (2018) Problem Solving in Data Structures & Algorithms Using C. Independently Published
8. Steven S. Skiena.(2020) The Algorithm Design Manual. 3rd ed. Cham: Springer Nature Switzerland AG
9. Allen Downey.(2017) Think Data Structures O'Reilly Media, Inc, USA:
10. Шматко О. В., Стратієнко Н. К., Бородіна І. О.(2016) Методичні вказівки до виконання курсової роботи по курсу "Алгоритми та структури даних" : для студ., які за напрямком 6.050103 "Програмна інженерія" спец. 05010301 "Програмне забезпечення систем" [Електронний ресурс] / уклад. Харківський політехнічний ін-т, нац. техн. ун-т. – Електрон. текстові дані. Харків: Retryeved from: http://repository.kpi.kharkov.ua/handle/KhPI-Press/24697.
11. Shmatko, O. V., Stratiienko N. K., Borodina I. O. (2016) 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". Electronic text data. Kharkiv, Access mode: Retryeved from http://repository.kpi.kharkov.ua/handle/KhPI-Press/24695.

Assessment and grading

Criteria for assessment of student performance, and the final score structure

100% final assessment in the form of a test (30%) and a current assessment (70%).
30% credit: semester credit, according to the schedule of the educational process
70% current assessment:
- 18% assessment of tasks in laboratory work;
- 22% assessment of the calculation task;
- 30% intermediate control (2 module tests).

Grading scale

Total points	National	ECTS
90–100	Excellent	A
82–89	Good	B
75–81	Good	C
64–74	Satisfactory	D
60–63	Satisfactory	E
35–59	Unsatisfactory (requires additional learning)	FX
1–34	Unsatisfactory (requires repetition of the course)	F

Norms of academic integrity and course policy

The student must adhere to the Code of Ethics of Academic Relations and Integrity of NTU "KhPI": to demonstrate discipline, good manners, kindness, honesty, and responsibility. Conflict situations should be openly discussed in academic groups with a lecturer, and if it is impossible to resolve the conflict, they should be brought to the attention of the Institute's management.
Regulatory and legal documents related to the implementation of the principles of academic integrity at NTU "KhPI" are available on the website: http://blogs.kpi.kharkov.ua/v2/nv/akademichna-dobrochesnist/

Approval

Approved by 08.06.2023 Head of the department
Ihor HAMAIUN

08.06.2023 Guarantor of the educational program
Uliya LITVINOVA