



Syllabus

Course Program



Computer mathematics

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

5-6

Language of instruction

English, Ukrainian

Lecturers and course developers



Nataliia Khatsko

nataliia.khatsko@khpi.edu.ua

Academic degree, academic title, position

Ph.D., associate professor at the Department of Software Engineering and Management Information Technologies of NTU «KhPI», associate professor. Prepared and published more than 40 research papers and textbooks (SCOPUS Author ID

<https://www.scopus.com/authid/detail.uri?authorId=57200820629>;

Researcher ID

<https://app.webofknowledge.com/author/#/record/17252627>;

GoogleScholar

<https://scholar.google.com.ua/citations?user=US70vx4AAAAJ&hl=uk>; ORCID orcid.org/0000-0002-2543-0280)

Basic courses: "Computer Mathematics (parts 1, 2, 3)", "Practical seminar on mathematical methods in software engineering", "Formal methods of software verification", "Formal methods of software systems research"

[More about the lecturer on the department's website](#)

General information

Summary

Acquaintance of students with mathematical models and methodologies of operations research; study of general properties and solution of linear programming problems; . study of nonlinear optimisation methods, computer discrete mathematics.

Course objectives and goals

Formation of students' modern system of views in the field of computer mathematics, acquisition of practical skills in the use of formal methods and models of computer mathematics in information processing and description of processes related to software development; acquaintance of students with the basic concepts, models and methods of set theory, algebra of relations, mathematical logic, graph theory; formation of students' understanding of the basic models and methods of discrete mathematics,

acquaintance with the possibilities of applying the studied modes Formation of students' theoretical and practical knowledge of the theory of finite automata and its accompanying concepts (grammar, language, regular expression). Formation of students' modern system of views in the field of computer discrete mathematics, acquisition of practical skills in the use of formal methods and models of discrete mathematics in the processing of discrete information and description of discrete processes related to software development.

Format of classes

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

Competencies

K01. Ability to think abstractly, analyze and synthesize.
K05. Ability to learn and master modern knowledge.
K06. Ability to search, process and analyze information from various sources.
K20. Ability to apply fundamental and interdisciplinary knowledge to successfully solve software engineering problems.
K26. 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.
PLO05. To know and apply relevant mathematical concepts, methods of domain, system and object-oriented analysis and mathematical modeling for software development.
PLO18. To know and be able to apply information technologies for data processing, storage and transmission.

Student workload

The total volume of the course is 390 hours (13 ECTS credits): lectures - 80 hours, workshops - 96 hours, self-study - 214 hours.

Course prerequisites

Higher mathematics.

Theory of algorithms.

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

Teaching and learning methods:

interactive lectures with presentations, discussions, workshops, teamwork, case method, student feedback, problem-based learning.

Forms of assessment:

written individual assignments for workshops (CAS), assessment of knowledge in workshops (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 operations research

Topic 2. Form of writing and geometric interpretation of LP problems

Topic 3. Linear programming problem reference plans and their properties Sub-topics / list of questions

Topic 4. Method of sequential improvement of the plan (first algorithm)

Topic 5: The method of successive improvement of the plan (second algorithm) or the inverse matrix method.

Topic 6. The method of artificial basis
Topic 7. M-method for solving linear programming problems
Topic 8: Post-optimisation analysis in linear programming
Topic 9: General properties of integer programming problems
Topic 10. Methods for solving integer programming problems
Topic 11. Transport problems. Determination of the initial reference plan
Topic 12: Transport problems. Method of potentials
Topic 13. General properties of nonlinear programming problems
Topic 14: Numerical methods of one-dimensional unconditional optimisation
Topic 15. Numerical methods of the 0th order
Topic 16. Numerical methods of the 1st order
Topic 17. Numerical methods of the 2nd order
Topic 18: Conditional optimisation
Topic 19: Algebra of sets
Topic 20: Binary relations
Topic 21. Forms of representation and implementation of Boolean functions
Topic 22. The problem of minimising Boolean functions
Topic 23: Graphs. Basic concepts and definitions.
Topic 24. Reachability and connectivity in graphs
Topic 25. Colouring graphs
Topic 26. The tree. The skeleton of a graph
Topic 27: Algorithms for finding the shortest paths in a graph
Topic 28: Cycles and the traveling salesman problem
Topic 29: Streaming algorithms
Topic 30. Introduction to the theory of formal proofs
Topic 31: Alphabets, grammars and languages
Topic 32. Regular expressions, languages.
Topic 33. Deterministic finite automata
Topic 34. Non-deterministic finite automata
Topic 35. Finite automata with epsilon transitions
Topic 36. Finite automata and regular expressions
Topic 37. Converting a regular expression to a finite state machine
Topic 38. Automata with store memory

Topics of the workshops

Topic 1: Mathematical models and methodology of operations research. Construction of mathematical models.
Topic 2. Reference plans of linear programming problems and their properties. Solving LP problems based on the theorem of the existence of a reference plan.
Topic 3. Finite methods for solving LP problems. The method of sequential plan improvement
Topic 4. Artificial basis method and M-method.
Topic 5. Elements of duality theory in linear programming.
Topic 6: Integer linear programming. Gomori method.
Topic 7. The method of branches and boundaries.
Topic 8: Transport problems. Northwest corner method. The method of potentials.
Topic 9: Algebra of sets
Topic 10. Binary relations
Topic 11. Forms of representation and implementation of Boolean functions
Topic 12: Post classes
Topic 13. The problem of minimising Boolean functions
Topic 14: Basic concepts of graph theory.
Topic 15. Matrix problem of a graph
Topic 16. Operations on graphs
Topic 17. Reachability and connectivity in graphs.
Topic 18: Tree. The skeleton of the graph.
Topic 19: Algorithms for finding the shortest paths in a graph

- Topic 20: Hamiltonian paths, contours and the traveling salesman problem.
- Topic 21: Streaming algorithms.
- Topic 22. Deductive methods of proof. Construction of proofs.
- Topic 23. Solving problems of chain inference by inference rules, parsing chains
- Topic 24. Solving problems. Description of deterministic finite automata.
- Topic 25. Solving problems. Description of non-deterministic finite automata and their transformations.
- Topic 26. Transformations of automata
- Topic 27: Transforming a deterministic finite state machine into a regular expression.
- Topic 28. Converting a regular expression to a finite state machine..

Topics of the laboratory classes

The course does not include laboratory classes.

Self-study

Information on self-study and individual assignments (reports, course projects, etc.), if it is necessary according to the plan. Also, methods of control and assessment of self-study. The curriculum includes the completion of coursework (CW). At the beginning of the semester, students choose the topics of the course work from the list or propose their own topics and agree them with the teacher. The CW is completed during the semester and is defended during the test week or examination session. Students are recommended with additional materials (videos, articles) for self-study and processing.

Course materials and recommended reading

Compulsory materials

1. Hamdy A. Taha. (2017). Operations Research: An I Introduction (10th Global Edition). Pearson.
2. Godlevsky, M. D., Lisitsky, V. L., Stratienko, N. K. (2016). Operations Research: Problem Solving and Variants of Typical Calculations: Textbook for Students in Computer Science. Kharkiv : NTU KhPI.
3. Wayne L. Winston. (2021). Operations Research: Applications and Algorithms. (4th ed.). Cham: Springer Nature Switzerland AG.
4. Guzhva, V. O., Stratienko, N. K., Borodina, I. O. (2018). Methodical instructions for performing laboratory work in the course "Operations Research": [Electronic resource]: for students studying in speciality 121 "Software Engineering" and 122 "Computer Science". Kharkiv: NTU "KHPI".
5. John Vince. (2020). Foundation Mathematics for Computer Science: A Visual Approach. (2nd ed.). Springer.
6. Frederick S Hillier, Gerald J Lieberman. (2021). Introduction to operations research. (Eleventh ed.). New York, NY: McGraw-Hill EducG.
7. Ajit Singh (2019) Formal Language And Automata Theory.
8. Abejide Ade-Ibijola (2017) New Finite Automata Applications in Novice Program Comprehension. LAP LAMBERT Academic Publishing, 2017.
9. Neeru Gupta (2020) Beginner's Guide - Automata Theory.
10. Ezhilarasu Umadevi Palani (2019) Finite Automata Problems & Solutions. LAP Lambert Academic Publishing.
11. Stoyan Mihov, Klaus U. Schulz (2019) Finite-State Techniques: Automata, Transducers and Bimachines. - Cambridge University Press.

Additional materials

12. Jun Wu. (2018). The Beauty of Mathematics in Computer Science. Chapman & Hall.
13. Eric Lehman, F. Thomson Leighton, Albert R. Meyer. (2017). Mathematics for Computer Science. 12th Media Services.

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:

- 20% assessment of tasks in practical classes;
- 20% 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