Національний технічний університет України
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

Department of computer systems software

COMPUTATIONAL METHODS OF OPTIMIZATION

Working program of the academic discipline (Syllabus)

Details of the academic discipline	
Level of Higher Education	***First (bachelor's)***
Field of Study	*12 Information Technologies*
Specialty	*121 Software Engineering*
Education Program	*Software Engineering of Multimedia and Information Retrieval Systems*
Type of Course	*Selective*
Mode of Studies	*full-time*
Year of training, semester	*3rd year, autumn semester*
ECTS workload	*Lectures: 36 hours, computer workshop : 18 hours, independent work: 66 hours.*
Testing and assessment	*Assessment, modular control work, calendar control*
Course Schedule	*According to the schedule for the spring semester of the current academic year (rozklad.kpi.ua)*
Language of Instruction	*English*
Course Instructors	Lecturer: *Ph.D., Associate Professor, Onai Mykola* Practical training: *Ph.D., Associate Professor, Onai Mykola*

Outline of the Course

1. Course description, goals, objectives, and learning outcomes

Studying the discipline "Computational Methods of Optimization" allows students of higher education to develop the competencies necessary for quantitative justification of technical and managerial decisions based on the use of optimization methods.

***The goal** of studying the discipline "Computational Methods of Optimization" is the formation of students' ability to carry out innovative activities related to the development of software optimization methods used in decision support systems.*

***The subject of** the discipline "Computational Methods of Optimization" is software optimization methods.*

*The study of the discipline "Computational Methods of Optimization" strengthens the formation of students **of professional competences (PC)** necessary for solving practical tasks of professional activity:*
***PC08** Ability to apply fundamental and interdisciplinary knowledge to successfully solve software engineering problems.*
***PC15** Ability to apply fundamental and interdisciplinary knowledge to build advanced retrieval algorithms.*

*The study of the discipline "Computational Methods of Optimization" contributes to the formation in students of the following **program learning outcomes** (PLO) according to the educational program:*
***PLO01** To analyze, purposefully search for and select for the information and reference resources and knowledge necessary for solving professional tasks, taking into account modern achievements of science and technology.*

***PLO25** To know and to be able to use fundamental mathematical tools to build algorithms and develop modern software.*

2. Prerequisites and post-requisites of the course (the place of the course in the structural-logical scheme of studies in accordance with educational program)

The successful study of the discipline "Computational Methods of Optimization" is preceded by the study of the disciplines "Algorithms and data structures" and "Algorithmic support of multimedia and information-search systems" of the curriculum for bachelors in the specialty 121 Software engineering.

The theoretical knowledge and practical skills obtained as a result of mastering the discipline "Computational methods of optimization" can be useful for conducting scientific research and for completing bachelor's qualification work.

3. Content of the course

The discipline "Computational Methods of Optimization" involves the study of topics:

Topic 1. Methods of zero order

Topic 2. Methods of the first order

Topic 3. Methods of the second order

Topic 4. Genetic algorithms

Modular control work

Test

4. Educational materials and resources

Basic literature:

1. Nikulina O.M. Chyselni metody modeliuvannia ta optymizatsii upravlinnia dynamichnymy systemamy : navch. posibnyk z kursu «Chyselni metody» / O. M. Nikulina, V. P. Severyn. – Kharkiv : NTU «KhPI», 2024. – 144 p.
Use to study the principles of solving mathematical problems that arise during the construction of mathematical models. The materials are freely available on the Internet.

Additional literature:

2. William H. Press, Saul A. Teukolsky, William T. Vetterling, Brian P. Flannery Numerical Recipes 3rd Edition: The Art of Scientific Computing 3rd Edition.

Use to master practical skills in the discipline.

3. Mykel Kochenderfer, Tim Wheeler Algorithms for Optimization / Mykel Kochenderfer, Tim Wheeler // Springer, 2019.

Use to master the theoretical material of the discipline.

4. Xin-She Yang, Slawomir Koziel Computational Optimization and Applications in Engineering and Industry / Xin-She Yang, Slawomir Koziel // Springer, 2013

Use to master practical skills in the discipline.

5. Slawomir Koziel and Xin-She Yang Computational Optimization, Methods and Algorithms [Electronic resource], 2011. Access mode: https://doc.lagout.org/science/0_Computer%20Science/2_Algorithms/Computational%20Optimization%2C%20Methods %20and%20Algorithms%20%5BKoziel%20%26%20Yang%202011-06-17%5D.pdf

Use to master the principles of applying zero-order methods. The materials are freely available on the Internet.

Educational content

5. Methodology of mastering the discipline (educational component)

No.	Type of training session	Description of the training session
		Topic 1. Methods of zero order
1	Lecture 1. Basic provisions of computational methods of one-dimensional optimization	Svenn's algorithm. Models of optimization problems
2	Lecture 2. The method of dividing the interval in half	Unconditional optimization problems. Methods of calculating the ends of intervals.
3	Lecture 3 . Dichotomy method	Tasks unconditional optimization . Differences method dichotomies from method division in half solution _ _ optimization tasks
4	Computer workshop 1	Task: To develop a software system for solving optimization problems using the method of division in half or the method of dichotomy (depending on the option).
5	Lecture 4 . Golden ratio method	Tasks unconditional optimization . Differences method golden intersection from classical methods solution _ _ optimization tasks
6	Lecture 5 . Fibonacci method	Tasks unconditional optimization . Differences method Fibonacci from method golden intersection solution _ _ optimization tasks
7	Computer workshop 2	Task: Develop a software system for solving optimization problems using the golden ratio method or the Fibonacci method (depending on the option).
8	Lecture 6 . Rosenbrock method	Tasks unconditional optimization for functions several variables . Peculiarities of using the Rosenbrock method
9	Computer workshop 3	Task: To develop a software system for solving optimization problems using the Rosenbrock method.
		Topic 2. Methods of the first order
10	Lecture 7 . Gradient descent method	Expansion of an arbitrary function of several variables into a Taylor series. Peculiarities of finding a global extremum among a set of local extrema. Gradient descent method
11	Computer workshop 4	Task: Develop software for solving optimization problems using the gradient descent method.
12	Lecture 8 . The fastest gradient descent method	Application of numerical methods of zero-order one-dimensional optimization to recalculate the step at each iteration. Algorithm for implementing the fastest gradient descent method
13	Lecture 9 . Method of coordinate descent	A descent strategy for each coordinate of a function of several variables. Algorithm for implementation of the coordinate descent method

14	*Computer workshop 5*	*Task: Develop software for solving optimization problems using the fastest gradient descent method or coordinate descent method (depending on the option).*
Topic 3. Methods of the second order		
15	*Lecture 10. Newton's method*	*Newton's method for a function of one variable. Generalization to the n - dimensional case. The algorithm for implementing Newton's method for a function of several variables*
16	*Lecture 11. The Newton-Raphson method*	*Generalization of optimization methods using the second derivative function. The implementation algorithm of the Newton-Raphson method*
17	*Computer workshop 6*	*Task: Develop software for solving optimization problems using the Newton descent method or the Newton-Raphson method (depending on the option).*
18	*Lecture 12*	*Modular control work. Part 1*
Topic 4. Genetic algorithms		
19	*Lecture 13. General provisions, selection, crossing*	*General ideas of genetic algorithms. Types of genetic algorithms. Selection of data structures for the implementation of genetic algorithms. Methods of selection and crossing*
20	*Computer workshop 7*	*Task: Develop software components for selection and crossing.*
21	*Lecture 14. Mutation, selection of a new population*	*Methods of implementation of mutation and selection of a new population*
22	*Computer workshop 8*	*Task: Develop software components for selection and crossing.*
Topic 5. Methods of stochastic programming		
23	*Lecture 15. General statement of the problem of stochastic programming*	*Direct and indirect methods of solving stochastic programming problems. Examples of stochastic economic problems*
24	*Lecture 16. Peculiarities of the mathematical formulation of stochastic programming problems*	*P - models, M - models, D - models, types of restrictions, examples of problems (the problem of preserving assets, the problem of dividing areas)*
25	*Lecture 17. One-stage and two-stage problem of stochastic programming*	*Peculiarities of the formulation and methods of solving one -stage and two-stage stochastic programming problems*
26	*Computer workshop 9*	*Summing up.*
27	*Lecture 18*	*Modular control work. Part 2*

6. Independent work of a student/graduate student

The discipline "Computational Methods of Optimization" is based on independent preparation for classroom classes on theoretical and practical topics.

No. z/p	*The name of the topic submitted for independent processing*	*Number of hours*	*literature*
1	*Preparation for lectures*	*16*	*1-5*
2	*Preparation for a computer workshop*	*27*	*1-5*
3	*Preparation for modular control work. Part 1*	*9*	*1-5*
4	*Preparation for modular control work. Part 2*	*9*	*1-5*
5	*Preparation for the test*	*5*	*1-5*

Policy and Assessment

7. Policy of academic discipline (educational component)

Attending classes. *Absence from a classroom session does not involve the calculation of penalty points, since the student's final rating score is formed solely on the basis of the evaluation of study results. At the same time, discussion of the results of the thematic tasks, as well as presentation / public speaking and participation in discussions and additions at seminars will be evaluated during classroom classes. In order to actively participate in the work of the seminar, the student prepares for a specific seminar class in literature as recommended by the teacher. Participation in the work of the seminar also involves the preparation of reports and co-reports within all classes.*

Missed evaluation control measures. *Every student has the right to make up lessons missed for a valid reason (hospital, mobility, etc.) at the expense of independent work. More details at the link: https://kpi.ua/files/n3277.pdf .*

The procedure for contesting the results of assessment control measures. *A student may raise any issue relating to the assessment procedure and expect it to be dealt with in accordance with pre-defined procedures. Students have the right to challenge the results of control measures with arguments, explaining which criteria they disagree with according to the evaluation. Calendar control is carried out in order to improve the quality of students' education and monitor the student's fulfillment of the syllabus requirements.*

Academic integrity. *The policy and principles of academic integrity are defined in Chapter 3 of the Code of Honor of the National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute". More details: https://kpi.ua/code .*

Norms of ethical behavior. *Standards of ethical behavior of students and employees are defined in Chapter 2 of the Code of Honor of the National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute". More details: https://kpi.ua/code .*

Inclusive education. *The acquisition of knowledge and skills in the course of studying the discipline "Research activity in computer engineering" can be accessible to most people with special educational needs, except for students with serious visual impairments that do not allow them to perform tasks with the help of personal computers, laptops and/or other technical means.*

Studying in a foreign language. *In the course of the tasks, students may be recommended to refer to English-language sources. Assigning incentive and penalty points According to the Regulation on the system of evaluation of learning results, the sum of all incentive points cannot exceed 10% of the rating scale.*

All students must attend lectures and practical classes, where you need to actively work on learning the learning material. For objective reasons (for example - illness, international internship), training can take place in an online form individually upon agreement with the head of the course.

Deadlines and Rescheduling Policy:

Works that are submitted late without good reason will be assigned a lower grade. Rearranging modules takes place with the permission of the dean's office if there are good reasons (for example, sick leave).

***Policy on academic integrity** :*

All written works are checked for plagiarism and accepted for defense with correct textual borrowings of no more than 20%. Write-offs during control work are prohibited (including using mobile devices).

8. Types of control and rating system of assessment of learning outcomes

During the semester, students perform 8 computer workshops. The maximum number of points for each computer workshop: 6 points.

Points are awarded for:
- quality of performance of the computer workshop: 0-2 points;
- answer to theoretical questions during the defense of the computer workshop: 0-2 points;
- timely presentation of work for defense: 0-2 points.

Performance evaluation criteria:
2 points – the work is done qualitatively, in full;
1 point - the work is completed in full, but contains minor errors;
0 points – the work is incomplete or contains significant errors.

Answer evaluation criteria:
2 points – the answer is complete, well-argued;
1 point – the answer is generally correct, but has flaws or minor errors;
0 points - there is no answer or the answer is incorrect.

Criteria for evaluating the timeliness of work submission for defense:
2 points – the work is presented for defense no later than the specified deadline;
0 points – the work is submitted for defense later than the specified deadline.

The maximum number of points for performing and defending computer practicals:
6 points × 8 comp. practice = 48 points.

*The assignment for **the modular test** consists of 3 questions - 1 theoretical and 2 practical. The answer to a theoretical question is worth 6 points, and the answer to a practical question is worth 10 points.*

Evaluation criteria for each theoretical test question:
6 points – the answer is correct, complete, well-argued;
5 points – the answer is correct, detailed, but not very well argued;
4 points - in general, the answer is correct, but has shortcomings;
3 points – there are minor errors in the answer;
1-2 points – there are significant errors in the answer;
0 points - there is no answer or the answer is incorrect.

Evaluation criteria for the practical test question:
9-10 points – the answer is correct, the calculations are completed in full;
7-8 points - the answer is correct, but not very well supported by calculations;
5-6 points - in general, the answer is correct, but has flaws;
3-4 points – there are minor errors in the answer;
1-2 points – there are significant errors in the answer;

0 points - there is no answer or the answer is incorrect.

The maximum number of points for a modular control work:
*2 papers * (6 points × 1 theoretical question + 10 points × 2 practical questions) = 52 points.*

The rating scale for the discipline is equal to:
$R_c = R_{com.practice} + R_{MKR} = 48$ points + 52 points = 100 points.

Calendar control: is carried out twice a semester as a monitoring of the current state of fulfillment of the syllabus requirements.
At the first certification (7th week), the student receives "passed" if his current rating is at least 50% of the maximum number of points (20 points) that the student can receive before the first certification.
At the second certification (13th week), the student receives "passed" if his current rating is at least 50% of the maximum number of points (35 points) that the student can receive before the second certification.

Semester control: ***assessment***

Conditions for admission to semester control:
With a semester rating (R_C) of at least 60 points and the enrollment of all computer practical work, the graduate student receives credit "automatically" according to the table (Table of correspondence of rating points to grades on the university scale). Otherwise, he has to complete the credit control work.

Completion and protection of a computer workshop is a necessary condition for admission to the performance of credit control work.

A graduate student can try to improve his grade by writing a graded test, and his semester marks will be canceled ("hard" grading system).

The composition and evaluation criteria of the assessment test:

The test task *consists of 4 questions - 2 theoretical and 2 practical. The answer to each theoretical and practical question is evaluated by 25 points.*

Evaluation criteria for each theoretical test question:
24-25 points – the answer is correct, complete, well-argued;
21-23 points – the answer is correct, detailed, but not very well argued;
17-20 points - in general, the answer is correct, but has flaws;
12-16 points – there are minor errors in the answer;
1-11 points – there are significant errors in the answer;
0 points - there is no answer or the answer is incorrect.

Evaluation criteria for the practical test question:
24-25 points – the answer is correct, the calculations are completed in full;
21-23 points - the answer is correct, but not very well supported by calculations;
17-20 points - in general, the answer is correct, but has flaws;
12-16 points – there are minor errors in the answer;
1-11 points – there are significant errors in the answer;
0 points - there is no answer or the answer is incorrect.

The maximum number of points for a modular control work:
25 points × 2 theoretical questions + 25 points × 2 practical questions = 100 points.

Table of correspondence of rating points to grades on the university scale :

Scores	Grade
100-95	Excellent
94-85	Very good
84-75	Good
74-65	Satisfactory
64-60	Sufficient
Less than 60	Fail

Admission conditions not met	Not Graded

9. Additional information on the discipline (educational component)

The list of questions submitted for semester control is given to students in the last lesson.

Work program of the academic discipline (syllabus):

Is designed by Ph.D., Assoc. Prof., Onai M.V.

Adopted by Computer Systems Software Department (protocol № 8, 22 January 2025)

Approved by the Methodical commission of the Faculty of Applied Mathematics (protocol № 8, 03 February 2025)