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

Computer Systems Software Department

MATRIX CALCULUS SOFTWARE METHODS

Working program of the academic discipline (Syllabus)

Details of the academic discipline

Level of Higher Education	***Third (PhD)***
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 studies, semester	*2nd year, spring semester*
ECTS workload	*Lectures: 18 hours, laboratory classes: 18 hours, independent work: 114 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

The study of the discipline "Matrix Calculus Software Methods" allows students of higher education to develop the competencies necessary for solving complex problems of professional activity related to the development of software systems for solving typical problems that arise in the mathematical modeling of natural processes and phenomena.

The purpose *of studying the discipline "Matrix Calculus Software Methods" is the formation of students' ability to conduct scientific and innovative activities related to the development of software systems for performing matrix calculations.*

The subject of *the discipline "Matrix Calculus Software Methods" are methods of performing software calculations on matrices that have a special structure and large dimensions.*

The study of the discipline "Matrix Calculus Software Methods" strengthens the formation of students ***of general competencies (GC)****:*

- *The ability to search, process and analyze information from various sources.*
- *The ability to work in a team, to form positive relationships with colleagues, to communicate with the wider scientific community and the public in the field of software engineering.*

The study of the discipline "Matrix Calculus Software Methods" strengthens the formation of students ***of professional competencies (PC)*** *necessary for solving practical tasks of professional activity:*

- *The ability to obtain new scientific results that create new knowledge and make an original contribution to the development of software engineering and related interdisciplinary areas.*
- *The ability to conduct experimental studies to evaluate the efficiency and safety of software.*

Studying the discipline "Matrix Calculus Software Methods" contributes to students' formation of the following ***program learning outcomes*** *(PLO) according to the educational program:*

- *To plan and carry out experimental and/or theoretical research in software engineering and related interdisciplinary areas using modern tools and observing the norms of academic and professional ethics, critically analyze the results of own research and the results of other researchers in the context of the entire complex of modern knowledge regarding the problem under study.*
- *To formulate and test hypotheses; use appropriate evidence to substantiate the conclusions, in particular, the results of theoretical analysis, experimental studies and mathematical and/or computer modeling, available literature data.*

2. Pre-requisites and post-requisites of the discipline (place in the structural and logical scheme of training according to the relevant educational program)

The successful study of the discipline "Matrix Calculus Software Methods" is preceded by the study of the basic disciplines of the curriculum for the preparation of bachelors and masters of specialties belonging to the field of knowledge 12 Information technologies.

The theoretical knowledge and practical skills obtained as a result of mastering the discipline "Matrix Calculus Software Methods" theoretical knowledge and practical skills can be useful for conducting scientific research and preparing a dissertation.

3. Content of the academic discipline

The discipline "Matrix Calculus Software Methods" involves the study of topics:

Topic 1. Methods of solving matrix problems of eigenvalues

Topic 2. Singular decomposition of rectangular matrices

Topic 3. Software methods for solving systems of nonlinear equations

Topic 4. Approximation of functions

Modular control work

Test

4. Educational materials and resources

Basic literature:

1. Andrunyk V.A. Numerical methods in computer sciences: textbook / Andrunyk V.A., Vysotska V.A., Pasichnyk V.V., Chirun L.B., Chirun L.V. Ch // Volume 2 edited by V.V. Pasichnyka - Lviv: Novy Svit Publishing House, 2020. - 536 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 Numerical Recipes in C. The Art of Scientific Computing / William H. Press, Saul A. Teukolsky, William T. Vetterling, Brian P. Flannery // Cambridge University Press. - 1018 p.
Use to master practical skills in the discipline.
3. Walter Gautschi Numerical Analysis [Electronic resource], 2012. Access mode: http://www.ikiu.ac.ir/public-files/profiles/items/090ad_1410599906.pdf
Use to master the theoretical material of the discipline.
4. Singular Value Decomposition [Electronic resource] Access mode: https://www.cs.cmu.edu/~venkatg/teaching/CStheory-infoage/book-chapter-4.pdf
Use to study the general structure of the singular matrix decomposition. The materials are freely available on the Internet.

5. *5. Finding the Roots of Non-linear Equations Numerically using Newton's Raphson Method by A New Mathematical Technique [Електронний ресурс], 2022.*

Режим доступу: https://www.researchgate.net/publication/359128644_Finding_the_Roots_of_Non-linear_Equations_Numerically_using_Newton's_Raphson_Method_by_A_New_Mathematical_Technique
Use to study the principles of solving nonlinear equations. The materials are freely available on the Internet.
6. Nonlinearsolve.jl: high-performance and robust solvers for systems of nonlinear equations in julia [Електронний ресурс], 2024. Режим доступу: https://arxiv.org/pdf/2403.16341

Use to master practical skills in the discipline.

Educational content

5. Methodology of mastering the discipline (educational component)

No.	*Type of training session*	*Description of the training session*
Topic 1. Methods of solving matrix problems of eigenvalues		
1	*Lecture 1. The simplest methods of solving the partial problem of eigenvalues. Improved methods for solving the partial eigenvalue problem*	*Eigenpairs of matrices and their simplest properties. Problems reduced to the algebraic problem of eigenvalues. Rayleigh's relation. A step-by-step method for solving the partial problem of eigenvalues. Trace method.* Δ^2 *- the Aitken process.* *The method of scalar products for solving the partial eigenvalue problem. Rayleigh particle method. The method of reverse iterations. The method of reverse iterations with a shift. The method of inverse iterations with Rayleigh relations. Determination of the order of convergence of improved methods for solving the partial problem of roll values.* *Task on SRS: to consider examples of solving the partial problem of eigenvalues by the power method and its modifications, to sollve No. 1, to consider examples of solving the partial problem of eigenvalues by the method of scalar products, the method of inverse iterations and their modifications, to solve No. 2.*
2	*Computer workshop 1*	*Task: To develop a software system for solving matrix problems of eigenvalues.*
3	*Lecture 2 . The rotation method for solving the complete eigenvalue problem or the Jacobi method. LU-method of solving the complete asymmetric eigenvalue problem and its modification*	*Similarity transformation. Classical Jacobi method. Cyclic Jacobi method with a barrier. Analysis of key element selection strategies in the Jacobi method. The stopping criterion of the iterative process in the Jacobi method and its modifications. Application* LU *- decomposition of the matrix for the problem of finding all eigenpairs of the matrix. Hesenberg matrices.* *Task on SRS: consider examples of solving the complete problem of eigenvalues by various algorithms implemented by the Jacobi method, to slolve No. 3, to build* LU *- an algorithm for finding proper pairs of symmetric positive definite matrices in the basis* U^TU *- and* LL^T *-decomposition of Kholetskyi, to sollve No. 4.*
4	*Lecture 3. QR-method of solving the complete asymmetric eigenvalue*	*Application* QR *matrix decomposition for solving asymmetric spectral algebraic problems. Givens transformation. Modifications*

	problem and its modification	of the classic QR algorithm. Concept QB - factorization. Landslides for Wilkinson. Implicit Q theorem . Two-step QR - Algorithm of Francis. Task on SRS: derive formulas for building a matrix $\mathbf{H}$, which is used in QB -factorization and solve number 5.
5	Computer workshop 2	Task: To develop a software system for solving matrix problems of eigenvalues.
Topic 2. Singular decomposition of rectangular matrices		
6	Lecture 4 . Basic terms and concepts, a general approach to the construction of a singular expansion. Bidiagonalization of a square matrix	Applied problems in which it is necessary to perform a singular expansion of a rectangular matrix. A generalization of the concept of an eigennumber is a singular number. Right and left singular vectors. Stages of finding the singular expansion of a rectangular matrix. Application of QR decomposition and Givens transformations for bidiagonalization of a square matrix. The prosecution process. Exhaustion procedure. Task on SRS: investigate the relationship between right and left singular vectors, solve number 6. Derive formulas for constructing Householder matrices $\mathbf{H}_1^{(i)}$ and $\mathbf{H}_2^{(j)}$, which are used to reduce the square matrix $\mathbf{A}$ to the upper and lower two-diagonal view and solve #7
7	Computer workshop 3	Task: Develop software components for performing singular decomposition.
8	Lecture 5. Singular decomposition of a bidiagonal matrix. Examples of application of singular decomposition in engineering problems of bidiagonal matrix	Modification of the QR-algorithm with implicit shifts for solving the complete eigenvalue problem for performing the singular expansion of a bidiagonal matrix. Peculiarities of applying the Givens and Householder transformation to obtain a singular expansion. Calculation of the absolute value of the determinant of a square matrix. Spectral number of conditioning. Finding the solution of a homogeneous SLAR. Pseudo solution of SLAR. Normal pseudo-solution of SLAR. Moore-Penrose matrix. Pseudo-inverse matrix. Task on SRS: derive formulas for calculating the shift parameter τ and formulas for the pursuit process in the lower bidiagonal matrix and solve number 8. To consider examples of finding the rank of a matrix, a pseudo-inverse matrix and the condition number of a rectangular matrix and solve No. 9
9	Computer workshop 4	Task: To develop software components for demonstrating examples of the application of singular decomposition of rectangular matrices.
Topic 3 . Methods of solving systems of nonlinear equations		
15	Lecture 6. Method of simple iterations. Newton's method of solving systems of	The geometric content of the system of nonlinear equations. The main stages of solving a system of nonlinear equations. The problem of a fixed point of a nonlinear mapping. The method of continuous

	nonlinear equations and its simplified version. Modifications of Newton's method of solving systems of nonlinear equations	*iterations for solving a system of nonlinear equations. A sufficient condition for the convergence of the method of simple iterations. Newton's method of solving a system of nonlinear equations in an implicit form. Newton's explicit method of solving a system of nonlinear equations. Newton's simplified method of solving a system of nonlinear equations. Newton's two-stage method of solving a system of nonlinear equations. Newton's methods with successive approximation of inverse matrices. The classic method of cutting.* *Task on SRS: to analyze the conditions of the theorem on the convergence of the method of simple iterations for solving systems of nonlinear equations and solve No. 10. To consider examples of finding the solution of SNR by Newton's method and Newton's simplified method and solve No. 11. Consider examples of solving SNR using modifications of Newton's method and solve No. 12*
11	*Computer workshop 5*	*Task: To develop software modules for solving systems of nonlinear equations.*
12	*Lecture 14. Broyden's secant method, Brown's method and the connection between the problem of solving a system of nonlinear equations and the problem of optimization*	*Ratio of sec. Broyden's method of incisions. Brown's method. General view of optimization problems. Formulation of the problem of solving a system of nonlinear equations as an optimization problem.* *Task on SRS: to consider examples of solving SNR using Broyden's secant method and Brown's method and solve No. 13.*
13	*Computer workshop 6*	*Task: To develop software modules for solving systems of nonlinear equations.*
	Topic 4 . Approximation of functions	
14	*Lecture 8. Finite difference interpolation formulas*	*The concept of finite difference. Finite difference order. Connection of finite differences with derivatives. Diagonal table of finite differences.* *Newton's first interpolation polynomial. Newton's second interpolation polynomial. Newton's interpolation formulas. Central interpolation formulas. The first Gaussian interpolation polynomial. The second Gaussian interpolation polynomial. Stirling's interpolation formula. Bessel's interpolation formula.* *Tasks on SRS: solve number 14.*
15	*Computer workshop 7*	*Task: Develop software components for finding approximate values of tabular functions.*
16	*Lecture 9. Principles of construction of interpolation formulas for unequally spaced nodes. Principles of constructing interpolation formulas for unequally spaced nodes*	*Concept of split difference. Divided difference order. Split Difference Table. Newton's first interpolation formula for unequally spaced nodes. Newton's second interpolation formula for unequally spaced nodes. Concept of split difference. Divided difference order. Split Difference Table. Newton's first interpolation formula for unequally spaced nodes. Newton's second interpolation formula for unequally spaced nodes.* *Tasks on SRS: solve number 15 and 16.*

17	*Computer workshop 8*	*Task: To develop software components for visualization of the graph of the approximated function.*
18	*Computer workshop 9*	*Modular test and Results*

6. Independent work of a student/graduate student

The discipline "Matrix Calculus Software Methods" 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*	*54*	*1-6*
2	*Preparation for a computer workshops*	*54*	*1-6*
3	*Preparation for the semester control*	*6*	*1-6*

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 for evaluating learning outcomes (RSO)

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 module and semester control is in Appendix 1

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)

Appendix 1. List of questions submitted for module and semester control

Modular control work No. 1 includes the following types of tasks:

1. *Perform a singular decomposition of the matrix (at the first stage, reduce the matrix to the upper bidiagonal form).*
2. *Perform a singular expansion of the matrix (at the first stage, reduce the matrix to the lower bidiagonal form)*
3. *Find all the eigenvalues and eigenvectors of the matrix with accuracy algorithm No. 0 for the implementation of the Jacobi method.* $\varepsilon \le 10^{-2}$ *If the required number of iterations exceeds the value of 3, then it is allowed to limit to three iterations.*
4. *Find all the eigenvalues and eigenvectors of the matrix with accuracy using algorithm No. 1 of the Jacobi method implementation.* $\varepsilon \le 10^{-2}$ *If the required number of iterations exceeds the value of 3, then it is allowed to limit to three iterations.*
5. *Find all the eigenvalues and eigenvectors of the matrix with accuracy using algorithm No. 2 of the Jacobi method implementation.* $\varepsilon \le 10^{-2}$ *If the required number of iterations exceeds the value of 3, then it is allowed to limit to three iterations.*
6. *LU-method (single diagonal in matrix L), with accuracy* $\varepsilon \le 10^{-2}$*, to find all the eigenvalues of the matrix. If the required number of iterations exceeds the value of 3, then it is allowed to limit to three iterations.*
7. *Using the LLT method, with accuracy* $\varepsilon \le 10^{-2}$*, find all the eigenvalues of the matrix. If the required number of iterations exceeds the value of 3, then it is allowed to limit to three iterations.*
8. *The UTU method is used* $\varepsilon \le 10^{-2}$ *to find all the eigenvalues of the matrix with accuracy . If the required number of iterations exceeds the value of 3, then it is allowed to limit to three iterations.*
9. *Algorithm No. 0 of the implementation of the QR method can find all eigenvalues of the matrix with accuracy .* $\varepsilon \le 10^{-2}$ *If the required number of iterations exceeds the value of 3, then it is allowed to limit to three iterations.*
10. *Algorithm No. 1 of implementing the QR method can find all the eigenvalues of the matrix with accuracy .* $\varepsilon \le 10^{-2}$ *If the required number of iterations exceeds the value of 3, then it is allowed to limit to three iterations.*
11. *Algorithm No. 2 of implementing the QR method can find all eigenvalues of the matrix with accuracy .* $\varepsilon \le 10^{-2}$ *If the required number of iterations exceeds the value of 3, then it is allowed to limit to three iterations.*
12. *Algorithm No. 3 of the implementation of the QR-method can be used to find all eigenvalues of the matrix with accuracy .* $\varepsilon \le 5 \cdot 10^{-2}$ *If the required number of iterations exceeds the value of 3, then it is allowed to limit to three iterations.*
13. *LU-method (unit diagonal in matrix U), with accuracy* $\varepsilon \le 10^{-2}$*, to find all the eigenvalues of the matrix. If the required number of iterations exceeds the value of 3, then it is allowed to limit to three iterations.*
14. *Algorithm No. 4 of implementing the QR-method can find all eigenvalues of the matrix with accuracy .* $\varepsilon \le 5 \cdot 10^{-2}$ *If the required number of iterations exceeds the value of 3, then it is allowed to limit to three iterations.*
15. *Algorithm No. 5 of implementing the QR method can find all eigenvalues of the matrix with accuracy .* $\varepsilon \le 5 \cdot 10^{-2}$ *If the required number of iterations exceeds the value of 3, then it is allowed to limit to three iterations.*

16. *Bring the given matrix to the Hesenberg form and find all eigenvalues and eigenvectors corresponding to them with the accuracy of Jacobi #0 method* $\varepsilon \le 10^{-2}$*. If the required number of iterations exceeds the value of 3, then it is allowed to limit to three iterations.*
17. *Bring the given matrix to the Hesenberg form and find all eigenvalues and eigenvectors corresponding to them with the accuracy of Jacobi method #1* $\varepsilon \le 10^{-2}$*. If the required number of iterations exceeds the value of 3, then it is allowed to limit to three iterations.*
18. *Bring the given matrix to the Hesenberg form and find all eigenvalues and eigenvectors corresponding to them with the accuracy of Jacobi method #2* $\varepsilon \le 3 \cdot 10^{-2}$*. If the required number of iterations exceeds the value of 3, then it is allowed to limit to three iterations.*
19. *Bring the given matrix to the Hesenberg form and find all eigenvalues and eigenvectors corresponding to them with the accuracy of Jacobi method #3* $\varepsilon \le 5 \cdot 10^{-2}$*. If the required number of iterations exceeds the value of 3, then it is allowed to limit to three iterations.*
20. *Find all the eigenvalues and eigenvectors of the matrix with accuracy using algorithm No. 4 of the Jacobi method implementation.* $\varepsilon \le 10^{-2}$ *If the required number of iterations exceeds the value of 3, then it is allowed to limit to three iterations.*
21. *Find all the eigenvalues and eigenvectors of the matrix with accuracy using algorithm No. 5 of the Jacobi method implementation.* $\varepsilon \le 7 \cdot 10^{-2}$ *If the required number of iterations exceeds the value of 3, then it is allowed to limit to three iterations.*
22. *with precision* $\varepsilon \le 10^{-2}$ *using any method (indicate the name of the chosen method). If the required number of iterations exceeds the value of 3, then it is allowed to limit to three iterations.*

Modular control work No. 2 includes the following types of tasks:

1. *Perform localization of all roots of the system of nonlinear equations. Find one of the localized roots by any iterative method of specifying the roots of the system of nonlinear equations (indicate the name of the selected iterative method) with accuracy* $\varepsilon \le 5 \cdot 10^{-2}$*. If the required number of iterations exceeds the value of 3, then it is allowed to limit to three iterations.*
2. *Perform localization of all roots of the system of nonlinear equations. Find one of the localized roots by Newton's explicit method of solving systems of nonlinear equations with accuracy* $\varepsilon \le 10^{-2}$*. If the required number of iterations exceeds the value of 3, then it is allowed to limit to three iterations.*
3. *Perform localization of all roots of the system of nonlinear equations. Find one of the localized roots by Newton's implicit method of solving systems of nonlinear equations with accuracy* $\varepsilon \le 10^{-1}$*. If the required number of iterations exceeds the value of 3, then it is allowed to limit to three iterations.*
4. *Perform localization of all roots of the system of nonlinear equations. Find one of the localized roots by Newton's simplified method of solving systems of nonlinear equations with accuracy* $\varepsilon \le 10^{-2}$*. If the required number of iterations exceeds the value of 3, then it is allowed to limit to three iterations.*
5. *Perform localization of all roots of the system of nonlinear equations. Find one of the localized roots by Newton's two-step method of solving systems of nonlinear equations with accuracy* $\varepsilon \le 7 \cdot 10^{-2}$*. If the required number of iterations exceeds the value of 3, then it is allowed to limit to three iterations.*
6. *Perform localization of all roots of the system of nonlinear equations. Find one of the localized roots by approximation analogue No. 1 of Newton's method of solving systems of nonlinear*

equations with accuracy $\varepsilon \le 10^{-2}$. If the required number of iterations exceeds the value of 3, then it is allowed to limit to three iterations.

7. *Perform localization of all roots of the system of nonlinear equations. Find one of the localized roots by approximation analog No. 2 of Newton's method of solving systems of nonlinear equations with accuracy $\varepsilon \le 10^{-2}$. If the required number of iterations exceeds the value of 3, then it is allowed to limit to three iterations.*
8. *Perform localization of all roots of the system of nonlinear equations. Find one of the localized roots by approximation analog No. 1 of Newton's two-stage method of solving systems of nonlinear equations with accuracy $\varepsilon \le 10^{-2}$. If the required number of iterations exceeds the value of 3, then it is allowed to limit to three iterations.*
9. *Perform localization of all roots of the system of nonlinear equations. Find one of the localized roots by approximation analog No. 2 of Newton's two-stage method of solving systems of nonlinear equations with accuracy $\varepsilon \le 10^{-2}$. If the required number of iterations exceeds the value of 3, then it is allowed to limit to three iterations.*
10. *Perform localization of all roots of the system of nonlinear equations. Find one of the localized roots by discrete (choose the discretization step constant for all iterations) Newton's explicit method of solving systems of nonlinear equations with accuracy $\varepsilon \le 3 \cdot 10^{-2}$. If the required number of iterations exceeds the value of 3, then it is allowed to limit to three iterations.*
11. *Perform localization of all roots of the system of nonlinear equations. Find one of the localized roots by discrete (choose the discretization step to be constant for all iterations) Newton's implicit method of solving systems of nonlinear equations with accuracy $\varepsilon \le 7 \cdot 10^{-2}$. If the required number of iterations exceeds the value of 3, then it is allowed to limit to three iterations.*