



Engineering calculations on a computer
Work program of the educational component (Syllabus)

Details of the educational component

Higher education level	<i>First (educational and professional)</i>
Field of Knowledge	<i>13 Mechanical Engineering</i>
Speciality	<i>131 Applied Mechanics</i>
Educational program	<i>Packaging and packaging equipment engineering</i>
Status of the educational component	<i>Mandatory</i>
Form of study	<i>full-time (full-time)</i>
Year of preparation, semester	<i>2nd year, 3rd semester</i>
Scope of the educational component	<i>4 cr. / 120 year. (72 room 48 SRS)</i>
Semester control / control measures	<i>Passed</i>
Class schedule	<i>4 hours per week (2 hours of lectures and 2 hours of laboratory classes)</i>
Language of instruction	<i>Ukrainian</i>
Information about the course leader / teachers	<i>http://intellect.cpsm.kpi.ua/profile/sde9</i>
Course placement	<i>Sikorsky Platform</i>

Program of the educational component

1. Description of the educational component, purpose, subject of study and learning outcomes

The student receives basic knowledge of engineering calculations on computers, rules for working with a computer, the basics of programming; will be able to perform calculations and evaluate the results obtained in software environments, MSDev FPS, Fortran, or similar.

The subject is engineering calculations using a programming language specialized for scientific and engineering calculations.

1.1. The purpose of the educational component is to participate in the formation of the following competencies in accordance with the requirements of the educational program.

Skills in the use of information and communication technologies (ZK 09).

Ability to use analytical and numerical mathematical methods to solve problems of applied mechanics, in particular, to perform calculations for strength, endurance, stability, durability, rigidity in the process of static and dynamic load in order to assess the reliability of parts and structures of machines (FC 05).

Ability to apply computerized design (CAD), manufacturing (CAM), engineering research (CAE) and specialized application software systems to solve engineering problems in applied mechanics (FC 07).

1.2. According to the requirements of the educational program, students, after mastering the educational component "Engineering calculations on a computer", must demonstrate the following program learning outcomes:

Know and understand the basics of information technology, programming, practically use application software to perform engineering calculations, process information and experimental research results (PRN 08).

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

The educational component "Engineering Calculations on Computers" is based on the educational component "Informatics". This educational component provides educational components "Technological processes and equipment. Interdisciplinary Course Work", "Fundamentals of CAD/CAM/CAE Application", Course and Diploma Design.

3. Content of the educational component

Introductory provisions. Rating system. Safety and ergonomics when working on a computer

Chapter 1. Operating system and the basics of computer functioning

Topic 1.1. Technical means of computers. Operating system. Team. File management.

Chapter 2. Fundamentals of programming in the algorithmic language Fortran

Topic 2.1. General information about the algorithmic language Fortran. Compiler file structure. Compile source code.

Topic 2.2. Algorithm. Types of algorithms. Drawing up flowcharts. Table of identifiers.

Topic 2.3. Algorithmic language Fortran. Programming Basics.

Chapter 3. Working with spreadsheets. IDE. MS Dev, FPS programming system. Procedural programming

Topic 3.1 Working with spreadsheets.

Topic 3.2. IDE. MS Dev environment and modern Fortran in FPS 4 implementation.

Topic 3.3. Program units

Topic 3.4. Data transfer between software units

Chapter 4. Modeling Basics

Topic 4.1. Modeling method

Topic 4.2. Models based on SDR and methods for their solution

4. Training materials and resources

Basic literature

1. Engineering calculations on computers: Tasks for independent solution. Teaching aids for the training of bachelors in the specialties 131 – Applied Mechanics, 133 – Branch Mechanical Engineering / Compiled by: D.E. Sidorov, I.O. Kazak. – Kyiv: I. Sikorsky Kyiv Polytechnic Institute, 2019. – 99 p. Access: <http://ela.kpi.ua/handle/123456789/27600>

2. Engineering Calculations on Computers. Part 1. Programming. Laboratory Practicum in the Academic Discipline: Teaching Aids for the Preparation of Full-Time Bachelors in the Specialty 133 – Branch Mechanical Engineering / Compiled by D.E. Sidorov, I.O. Kazak, I.I. Ivitsky. – Kyiv: I. Sikorsky Kyiv Polytechnic Institute, 2018. – 66 p. Access: <http://ela.kpi.ua/handle/123456789/25145>

3. Engineering calculations on computers. Part 2. Procedural programming. Laboratory workshop on the academic discipline: teaching aids for the training of full-time bachelors in the specialty 133 – Branch Mechanical Engineering / Compiled by D.E. Sidorov, I.O. Kazak, I.I. Ivitsky. – Kyiv: Igor Sikorsky Kyiv Polytechnic Institute, 2018. – 38 p. Access: <http://ela.kpi.ua/handle/123456789/25146>

4. D.E. Sidorov, and O. Kazak. Basics of engineering calculations on computers. Programming in the algorithmic language FORTRAN. – Kyiv: Center for Educational Literature, 2016. – 185 p. (in Russian).

5. Software of engineering calculations: Laboratory workshop [Electronic resource]: study. Manual. for the training of bachelors of specialty 133 "Industrial Engineering", educational program "Computer-integrated technologies for designing chemical engineering equipment" / KPI. Igor Sikorsky; compiled by: Sidorov D.E., I.O. Kazak. – Kyiv: KPI. Igor Sikorsky, 2022. – 166 p. (in Russian).

Further reading

1. Methodical instructions for performing tasks on laboratory work and independent work of students in the credit module "Engineering calculations on computers – 1. Programming" / compiled by: D.E. Sidorov, I.O. Kazak. – Kyiv: NTUU "KPI them. Igor Sikorsky", 2016. – 86 p. (in Russian).

2. *Methodical instructions for performing tasks on laboratory work and independent work of students on the credit module "Engineering calculations on a computer - 2. Procedural Programming and the Basics of Modeling" / Compiled by: D.E. Sidorov, I.O. Kazak. – K. NTUU "KPI them. Igor Sikorsky", 2016. – 48 p. (in Russian).*
3. *Informatics: Excel and Basic for Application: Manual. Manual. / Lopotko O.V. Recommended by the Ministry of Education and Science of Ukraine. Kyiv: Karavela Publ., 2018. 272 c.*
4. *Chapman, Stephen J. Fortran for scientists and engineers / Stephen J. Chapman, BAE Systems Australia. Fourth edition. | New York, NY : McGraw-Hill, a business unit of The McGraw-Hill Companies, Inc. (March 16, 2017) ISBN 978-0-07-338589-1; MHID 0-07-338589-1*
5. *Methodical instructions for the implementation of tasks from the computer workshop and independent work of students in the discipline "Engineering calculations on a computer – 1. Programming", for students of the areas of training 6.050503 "Mechanical Engineering", 6.050502 "Engineering Mechanics" / Compiled by: D.E. Sidorov, I.O. Kazak. – KYIV: NTUU "KPI", 2012. – 86 p. (in Russian).*
6. *Engineering calculations on personal electronic computers – 2. Procedural Programming and the Basics of Modeling: Methodical Instructions for Performing the Tasks of Computer Practicum and Independent Work of Students in the Areas of Training 6.050503 – Mechanical Engineering, 6.050502 – Engineering Mechanics / Compiled by: D.E. Sidorov, I.O. Kazak. – KYIV: NTUU "KPI", 2013. – 48 p. (in Russian).*

Information resources on the Internet

1. *The program of the educational component - in the "Campus" system and on the "Sikorsky" platform;*
2. *Work program of the educational component (syllabus) - in the "Campus" system, on the "Sikorsky" platform and on the website of the department (faculty);*
3. *Rating system of assessment in the "Campus" system on the "Sikorsky" platform;*
4. *Synopsis of lectures on the Sikorsky platform;*
5. *Methodological guidelines and teaching aids:*
<http://ela.kpi.ua/handle/123456789/27600>
<http://ela.kpi.ua/handle/123456789/25145>
<http://ela.kpi.ua/handle/123456789/25146>
<https://fortranwiki.org/fortran/show/Fortran+Courses>
<http://faculty.washington.edu/rjl/uwhpsc-coursera/fortran.html>
6. *Distance courses:*
<https://www.udemy.com/course/fortran-for-beginners/>
<https://siit.co/courses/fortran-course-and-certification/414>
<https://www.udemy.com/course/intermediate-fortran/>
<https://www.udemy.com/course/fortran-course/>
<https://www.classcentral.com/subject/fortran?certificate=true>
<https://www.classcentral.com/course/fortran-for-scientific-computing-56029>
<https://www.futurelearn.com/courses/fortran-for-scientific-computing>

5. Methods of mastering the educational component

Lectures (36 hours)

Lectures are aimed at:

- providing modern, holistic, interdependent knowledge on the educational component "Engineering calculations on a computer", the level of which is determined by the target setting for each specific topic;
- ensuring the creative work of bachelors together with the teacher during the lecture;
- education of bachelors' professional and business qualities and the development of independent creative thinking in them;
- formation of the necessary interest in bachelors and providing direction for independent work;
- reflection of methodological processing of the material (highlighting the main provisions, conclusions, recommendations, clear and adequate to their formulation);
- use of visual materials for demonstration, combining, if possible, them with the demonstration of results and samples;
- teaching the materials of the discipline in a clear and high-quality language in compliance with structural and logical connections, explaining all newly introduced terms and concepts;
- accessibility for perception by this audience.

No. s/p	Title of the lecture topic and list of main questions	Hours
1.	Introductory provisions. Rating system. Safety and ergonomics when working on a computer Objectives of the course, the role and place of discipline in the structure of the educational process. The use of computers in solving mathematical models. Modeling and engineering. Achievements in the field of calculations and modeling of technical objects. Literature recommended for study. Provisions of the rating system for assessing student performance in the implementation of the module program.	2
2.	Technical means of computers and OS Technical means of computers. BIOS. Self-diagnosis of the computer when turned on. OS. OS functions. OS classification, examples. OS program structure (examples of the simplest single-tasking MS-DOS and WIN).	2
3.	Working in the console File types. Console. OS commands. Automate work with batch command files. File management.	2
4.	Classification of translators and the city of Fortran in her Interpreters and compilers. High and low level languages. Algorithmic language Fortran, use in technical calculations, roots and prospects for development. The composition of the Fortran programming system with examples. Broadcast procedure. Obtaining *.obj and *.exe modules. Compile batch command files to control the compilation process.	2
5.	Algorithmic language Fortran. Rules for generating a source code file Algorithm, requirements for the algorithm. Flowchart of the representation of the algorithm Table of identifiers. Alphabet and syntax. Priorities of arithmetic operations. Fortran screen. Rules for writing operators. Executable and non-executable statements. Label, comments, hyphenation. Output in a simple form. The structure of the program, the simplest program. Operators PROGRAM, STOP, END. I/O in a simple form. I/O operators. Dialog mode of the program.	2
6.	Data type description operators Data types. Operators INTEGER, REAL, COMPLEX, CHARACTER, LOGICAL. Rules for working with data of the same type and different types. Assignment and default rules for data types. IMPLICIT operator. Giving a name to data: constants and variables,	2

	<i>PARAMETER, length name, assignment rules. Features that are standard. Examples.</i>	
7.	Conditional control operators. Branching <i>Control operators. Unconditional transition. Three types of conditional control operators. Arithmetic and logical conditional control operators. Boolean expression. Relationship. Logical actions.</i>	2
8.	Block conditional control operator. <i>Block conditional control operator. Determination of the logical level of the operator. Multivariate branching.</i>	2
9.	Cyclic calculations <i>Cycles. Organization of cyclical processes. Arrays. Static and dynamic data.</i>	2
10.	Formatted I/O* <i>Formatted I/O. Format specifiers for real and integer. Format specifiers for non-numeric data.</i>	(2*)
11.	Working with the computer code table* <i>Computer code table, pseudographics. Formation of a tabular output. Metacommands. Program code optimization. Search and work out errors of the "run-time error" type.</i>	(2*)
12.	Working with spreadsheets <i>Working with spreadsheets and building diagrams. Import and export data using text files.</i>	2
13.	IDE on the example of MS Dev and modern Fortran in the implementation of FPS 4 <i>Purpose of IDE, examples of IDEs, MS Dev. FPS capabilities 4. Working with visual tools for software development MS Dev. Project. Project files. BUILD menu. Compiling project files. DEBUG and RELEASE compilation modes. Icon editor.</i>	2
14.	Program units <i>Ideology of procedural programming. Program units. Broadcasting of program units.</i>	2
15.	Program units <i>Main program. Subroutine. Formal and actual parameters of the Function. Types of functions. Block date..</i>	2
16.	Data transfer between software units <i>Object visibility zone. Transfer data between software units through parameter lists. Internal procedures. Transfer data between software units using shared memory units.</i>	2
17.	Modules <i>Application of modules. Expanding the visibility zone of objects.</i>	2
18.	Working with files <i>Transfer data between software units using a file interface.</i>	2
19.	Modeling method <i>Model definition and modeling. Generalized diagram of the modeling process. Object, physical model, mathematical model, solution. Accuracy of solution for mathematical models.</i>	2
20.	Models based on SDR and methods for their solution <i>Structure of analytical and numerical solution of mathematical models. Resistance. Consistency. Convergence. Advantages of methods, disadvantages and features. Example.</i>	2
21.	Technical models* <i>Modeling the process of heating (cooling) the body. Description of the real process, formulation of a physical model based on hypotheses, mathematical model. Euler's method for solving ADR. Application of the method for numerical solution of a composite mathematical model.</i>	(2*)
22.	Chemical reaction models * <i>Generalized equation of a chemical reaction. The law of acting masses is the basis for building a mathematical model. Differential model. Runge-Kutta method for numerical solution of ZDR. Application of a method to solve a composite model.</i>	(2*)

23.	Differential models in ecology * <i>Objects of research in ecological models. Physical model and its clarification, mathematical model, obtaining a solution and its analysis. (The method of Euler, Runge-Kutt or M. K. R. for solving a composite model.)</i> Economic models * <i>Advertising effectiveness. Logistics curve. Exact and numerical solution. Supply and demand. Objects of study, physical models, mathematical models, solutions and analysis.</i>	(2*)
	Just	36

**-Optional*

Under the condition of blended learning technology (individual work 24 hours), it is proposed to teach the main lecture material (12 hours of classroom classes) in the review mode of presentation. Lectures 3, 5, 6-9 – if possible, in full.

Laboratory classes (36 hours)

The purpose of performing the tasks of laboratory work is to acquire skills and abilities in the use of computers in engineering calculations, to study the capabilities of the programming system for the algorithmization of engineering calculations. The student also works out the skills of working with a computer (the level of skill formation is the ability to perform an action automatically, at the skill level). The level of formation of skills in working with the relevant software products is the ability to perform an action, relying on constant mental control without the help of material media.

No. s/p	Job Title	Points	Number of auditoriums. Hours
1.	Introduction. Safety precautions. Introductory provisions.		2
2.	Technical means of computers. Operating system. Some operating system commands. File management.	8	4
3.	Work with the Fortran programming system.	4	2
4.	Input and output operations, calculation of arithmetic expressions	8	2
5.	Conditional control operators, unconditional transition operator	8	4
6.	Cyclic processes. Organizing loops using IF and DO statements	8	4
7.	Format output. Building tables. Working with the table of computer codes. Working with arrays*.	*	4*
8.	Working with spreadsheets. Diagramming	4	2
9.	Work with the MSDev programming system (FPS 4).	4	2
10.	Icon Editor. Development and application of icons	4	2
11.	Design of the project structure using the procedural paradigm	8	2
12.	Creating programs using subroutines and functions: - data transfer using parameter lists; - data transfer using shared memory units; - the use of modules; - file interface.	16	6
13.	Solution of differential models based on ZDR according to the Euler (Runge-Kutt) method	8	2
14.	Modular control work	20	2
15.	Saturn Software.*		2*
Just		100	36

** - optional, additional or optional works.*

6. Independent work of a student

Independent work takes 40% of the time to study the educational component, including preparation for the test. The main task of independent work of bachelors is to master knowledge that is not included in the list of lecture questions through a personal search for information, the formation of an active interest in a creative approach in educational work. In the process of independent work within the framework of the educational component, the bachelor must learn to analyze modern sources, educational literature, and use certificate distance courses.

No. s/p	Name of the topic submitted to the SRC	Number of CPC hours
1.	Topic 1. General information about the algorithmic language Fortran. Compiler file structure. Compilation of source code. Preparation for laboratory work. Librarian. Step-by-step compilation by console commands. Alternative libraries.	4
2.	Topic 2. IDE. MS Dev environment and modern Fortran in FPS 4 implementation MS Dev work with deleted files. Features of Fortran 90. Preparation for computer workshops. Software development for individual tasks. FPS 4 format editor. MS Dev work with deleted files. Debugging program code.	4
3.	Topic 3. Algorithm. Types of algorithms. Drawing up flowcharts. Table of identifiers Execution of engineering documentation when performing calculations on a computer. Images of blocks, alternative blocks, standard capabilities of text editors for the image of blocks of flowcharts.	4
4.	Topic 4. Algorithmic language Fortran. Fundamentals of programming Preparing for laboratory work. Data input and output operators. Double precision for standard data types. Changing the default rule. Programming console data input. Redirecting program results. Boolean expressions. Boolean relations and logical operations/ Loops with post- and prerequisites. Standard functions for working with arrays. Tasks and changing boundaries. Cross section of an array. Format specifiers for non-numeric data. Code tables of Cyrillic characters. Search and work out errors of the "run-time error" type.	8
5.	Topic 5. Procedural programming. Program units Software development for individual tasks.	4
6.	Topic 6. Data transfer between software units Ways to place program units in project files. Data subroutines. Preparation for computer workshops. Software development for individual practical tasks.	8
7.	Preparation for the MCR	4
8.	Preparation for the test	6
	Just	48

You can independently complete the program of distance courses (Information resources on the Internet).

7. Educational Component Policy

Rules for attending classes and behavior in the classroom

Attendance at classes is not a mandatory component of assessment. Bachelors are obliged to take an active part in the educational process, not to be late for classes and not to miss them without a good reason, not to interfere with the teacher's conduct, not to be distracted by actions not related to the educational process.

Rules for assigning incentive and penalty points

Incentive points can be awarded by the teacher only for the performance of creative (optional) work in the discipline.

Additional points can be obtained by additional completion of on-line specialized courses with the receipt of the appropriate certificate:

<https://www.udemy.com/course/fortran-for-beginners/>

<https://siit.co/courses/fortran-course-and-certification/414>

<https://www.udemy.com/course/intermediate-fortran/>

<https://www.udemy.com/course/fortran-course/>

<https://www.classcentral.com/subject/fortran?certificate=true>

<https://www.classcentral.com/course/fortran-for-scientific-computing-56029>

<https://www.futurelearn.com/courses/fortran-for-scientific-computing>

Penalty points within the educational component can be received for late completion or defense of laboratory work.

Deadline and retake policy

In case of debts or any force majeure circumstances, bachelors should contact the teacher through the available (provided by the teacher) communication channels to solve problematic issues and agree on an algorithm of actions for working out.

Academic Integrity Policy

Plagiarism and other forms of dishonest work are unacceptable. Plagiarism includes the absence of references when using printed and electronic materials, quotes, opinions of other authors. All tasks designed for individual performance must be performed personally by the student. Hints and cheating during writing tests, control, conducting classes are unacceptable; defense of laboratory work, passing the exam for another student; copying materials protected by the copyright system without the permission of the author of the work.

The policy and principles of academic integrity are defined in Section 3 of the Code of Honor of the National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute". Read more: <https://kpi.ua/code>

Academic Conduct and Ethics Policy

Students should be tolerant, respect the opinions of others, formulate objections in the correct form, and constructively maintain feedback in the classroom.

The norms of ethical behavior of students and employees are defined in Section 2 of the Code of Honor of the National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute". Read more: <https://kpi.ua/code>

8. Types of control and rating system for assessing learning outcomes (RSO)

Distribution of study time by types of classes and tasks:

Semester	Study time	Distribution of training hours	Control measures
----------	------------	--------------------------------	------------------

	Loans	Acad. H.	Lecture	Practical (seminar)	Lab. Rob.	SRS	FDM	RR	Semester control
3	4	120	36	-	36	48	1	–	Passed

Note: depending on the number of students in the group, the conversion factor of classroom hours to hours of individual work can be applied.

The student's rating in the discipline consists of the points that he receives for:

- *Or for completing tasks from the laboratory workshop and MCR (MCR can be carried out in the form of open tasks, surveys in the classroom, or tests (the Saturn system, or on the Sikorsky platform). Ranking points obtained during the semester.;*
- *Or a test work (can be carried out in the form of open tasks, or in the form of tests of the Saturn system, or tests on the Sikorsky platform); The test is carried out in the case when the student wants to increase the grade, or the semester rating is insufficient – in this case, the points of the semester rating are canceled.*
- *as well as (at the request of the student) for independent completion of alternative distance courses.*

Semester control is a test.

System of rating (weight) points and evaluation criteria

Performing tasks in laboratory classes and MCR.

*The weight score for each task is given in the **Laboratory Work** section.*

Criteria for evaluating the performance of the task

Completeness and signs of task completion	%
<i>The task is completed in full, all the main and additional data are given</i>	<i>95-100</i>
<i>Minor deficiencies that do not significantly affect the results</i>	<i>80-94</i>
<i>Untimely completion of the task, shortcomings the student corrects on his own</i>	<i>50-79</i>
<i>Untimely completion of the task, significant shortcomings</i>	<i>20-49</i>
<i>Poor-quality, untimely completion of the task, shortcomings are corrected by the student provided that he uses his own note, or with the help of a contributor</i>	<i>Up to 20</i>
<i>Failure to complete the task</i>	<i>0</i>

Answer on the test work.

The ticket contains 4 questions. The weight score for each question is 25 points.

Criteria for evaluating the answer

Completeness and signs of task completion	%
<i>The task is completed in full, all the main and additional data are given</i>	<i>95-100</i>
<i>Minor deficiencies that do not significantly affect the results</i>	<i>75-94</i>
<i>Incomplete answer, some secondary data is not given. Shortcomings that affect the result, but the student's actions are in the right direction. The student can correct the answer on his own for the sake of correct results.</i>	<i>50-74</i>
<i>Incomplete answer, some basic data is not given. Shortcomings that affect the result, but the student's actions are in the right direction. The student can correct the answer if he uses his own synopsis.</i>	<i>20-50</i>
<i>Poor quality, task performance. The answer contains significant errors. The result is wrong, but the student's actions are in the right</i>	<i>Up to 20</i>

<i>direction. The student cannot correct the results without additional help from the teacher and the use of the notes.</i>	
<i>Failure to complete the task, lack of response</i>	<i>0</i>

For independent completion of alternative distance courses and the provision of a certificate, up to 100 points are awarded (determined individually). The total number of points in the discipline cannot exceed 100 points.

At the first attestation, a student receives "enrolled" if his current rating is at least half of the maximum number of points possible at the time of the first attestation.

At the second attestation, a student receives "enrolled" if his current rating is at least half of the maximum number of points possible at the time of the second attestation.

*To obtain a score, the sum of all received rating points **R** is converted according to the table:*

<i>Score</i>	<i>Score</i>
<i>95...100</i>	<i>Perfectly</i>
<i>85...94</i>	<i>Very good</i>
<i>75...84</i>	<i>well</i>
<i>65...74</i>	<i>Satisfactory</i>
<i>60...64</i>	<i>enough</i>
<i>RD < 60</i>	<i>Disappointing</i>
<i>Not met the conditions of admission</i>	<i>not allowed</i>

9. Additional information on the educational component

<https://cpsm.kpi.ua/>

Approximate list of questions that can be submitted to the ICR and for offset

Give general information about the algorithmic language Fortran and classify it.

What is the purpose of LINK.EXE, LIB.EXE programs?

Specify the file composition of the Fortran MS F77 v3.31 compiler. Describe the purpose of each file.

Give the definition of the algorithm. What are the known types of algorithms? Give the rules for drawing up flowcharts.

Explain what happens when the source code is compiled? Give and describe the stages of compilation.

Explain the purpose and rules of installation: label, comments, hyphenation.

Give the composition of the Fortran FTN95 programming system. Describe the purpose of each file.

List the ways to organize cyclical processes. Give examples for each case.

Give a classification of translators. Interpreters and compilers – disadvantages and advantages.

Give ways to organize cycles using conditional control operators. Give examples for each method.

Write according to the rules of Fortran: (task is provided).

Give the composition of the Fortran MS F77 V5.0 programming system. Describe the purpose of each file.

Write a program that displays a greeting on the screen: Hello, World!

Describe the general procedure for translating the source code. Give examples.

Explain the purpose, installation and operation of Fortran operators: INTEGER, REAL, COMPLEX, CHARACTER, LOGICAL. Give examples for each of the cases.

What are compiler passes? What happens in this case? When is the source code syntax check performed?

What is an ID table, its purpose? Give examples.

What is a console? Describe the features of working in the console mode of the OS.

Give images of blocks of flowcharts, describe the standard capabilities of text editors on the image of blocks of flowcharts.

Describe the procedure for obtaining the *.obj and *.exe module. What are these modules for?

Give the alphabet and syntax of Fortran. What symbols are used when writing operators and which are used to generate data?

Give a classification of control operators. How is an unconditional transition performed?

Explain the syntax and operation of conditional control operators.

Prioritize arithmetic operations using Fortran.

How are cycles implemented? Give examples of the organization of cyclical processes.

Explain the need for data typing. Give the main types of data, examples of application and explain them.

Specify and explain assignment and default rules for data types. Give examples for each of the cases.

Mark up the Fortran screen for a fixed format for placing operators. Explain the location of the operators.

What is the default rule for data types? How is a default rule change done? Give examples.

Explain the purpose and application of the IMPLICIT statement and the PARAMETER statement. Give examples.

What is the dialog mode of the application? Give examples.

Give the definition of the array. Explain the use of static and dynamic data.

Specify the typical structure of the Fortran program.

What is a console? How to get it? Describe the features of working in console mode.

Specify data naming rules: constants and variables, name length, assignment rules.

What are standard Fortran functions? Give examples of functions that are standard and describe the features of the application.

Give and explain the rule for bringing data types. Give examples.

What does a source code developer require to use the F90 extension in the file name of the input code? Explain with an example.

What is the default rule for data types? How is a default rule change done? Give examples.

Explain the purpose and application of the DIMENSION statement. Give examples.

Describe the syntax and operation of a conditional arithmetic control operator. Give an example.

Specify data naming rules: constants and variables, name length, assignment rules.

Describe the syntax and operation of the conditional logical control operator. Give an example.

Describe the syntax and operation of the conditional block control operator. Give an example.

Give cycle operators, describe the work. Give examples.

Explain what the procedural paradigm is and how it is implemented in Fortran.

What is a subroutine, a subroutine function? Give the syntactic form of the record and explain the features?

What functions are distinguished in Fortran? Give examples and explain.

What is the area of sight of objects? How do I change the visibility zone of objects?

To give and explain the methods of data transfer between program units.

What software units are used in Fortran? Explain the features.

How do you work with files in the Fortran program?

Work program of the educational component (syllabus):

Compiled by Assoc. Prof., Ph.D., Sidorov D.E.

Approved by the Department of KPSM (Minutes No. 17 of 04.06.2025)

Approved by the ICF Methodological Commission (Minutes No. 11 dated 27.06.2025)