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

Department of computer systems software

COMPUTER LOGIC. Part 2. Computer Arithmetics

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	*Normative*
Mode of Studies	*Full-time*
Year of studies, semester	*3 year of study, 5th semester*
ECTS workload	*4.5 ECTS credits, 135 hours (36 – lectures, 9 – practical classes, 9 – laboratory classes, self-study – 81)*
Testing and assessment	*Exam, modular control work, home control work*
Course Schedule	*http://rozklad.kpi.ua/*
Field of Study	*English*
Course Instructors	Lecturer: *Ph.D., Associate Professor, Onai Mykola* Practical training: *Ph.D., Associate Professor, Onai Mykola*
Placement of the course	*Electronic campus of NTUU "KPI". Materials from the discipline "Computer logic".*

Outline of the Course

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

Educational discipline "Computer Logic. Part 2. Computer Arithmetic" is aimed at an in-depth study of the theoretical foundations of computer arithmetic and the creation of a formal-mathematical basis for the study of hardware-technical and software-algorithmic disciplines. Appropriate theoretical and practical training forms basic skills for independent solving of scientific and technical programming problems for digital computing equipment.

The purpose of *the discipline is to form students' ability to analyze the accuracy, speed and complexity of computing procedures at the level of microoperations, to rationally choose the forms of information presentation at the level of machine words and algorithms for performing operations, to perform the necessary calculations to obtain technical and economic characteristics of information processing structures of digital computing equipment.*

The subject *of the educational discipline is the basic machine forms of digital information presentation, the basics of the theory of numbering and the theory of coding in relation to the needs of computer technology, algorithms for performing basic arithmetic logical operations, hardware-oriented algorithms for calculating elementary functions, functionally-oriented algorithms for performing group and multi-place operations, algorithmic systems of introcomp computer-level application.*

Discipline "Computer Logic. Part 2. Computer Arithmetics" forms ***general (GC)*** *and* ***professional competences (PC)*** *in students :*

GC01 *Ability to abstract thinking, analysis and synthesis*

GC02 *Ability to apply knowledge in practical situations*

GC05 *Ability to learn and master modern knowledge.*

PC03 *Ability to develop architectures, modules and components of software systems.*

Program learning outcomes (PLO) *disciplines "Computer Logic. Part 2. Computer Arithmetics» according to the educational program:*

PLO01 *To analyze, purposefully search and select the necessary information and reference resources and knowledge to solve professional problems, taking into account modern advances in science and technology.*

PLO02 *To know the professional ethics code, understand the social significance and cultural aspects of software engineering and adhere to them in professional activities*

PLO05 *To know and apply relevant mathematical concepts, domain methods, system and object-oriented analysis and mathematical modeling for software development*

PLO10 *To conduct a pre-project survey of the subject area, system analysis of the design object.*

PLO13 *To know and apply methods of developing algorithms, designing software and data and knowledge structures.*

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)

Successful mastering of the discipline "Computer Logic. Part 2. Computer Arithmetics" precedes the study of the disciplines "Computer Logic. Part 1. Applied Theory of Digital Automata", "Mathematical Support of Multimedia and Information Retrieval Systems", "Mathematical Analysis", "Computer Discrete Mathematics", "Theory of Probability".

Received during the assimilation of the discipline "Computer Logic. Part 2. Computer Arithmetics" theoretical knowledge and practical skills ensure successful completion of pre-diploma practice, completion of course and diploma projects in the specialty 121 Software Engineering.

3. Content of the course

Chapter 1. Numerical systems in computer arithmetic
Topic 1.1. Computer arithmetic
Topic 1.2. General characteristics of operational resources of computers and computer systems
Topic 1.3. Computer aspects of the theory of calculation systems
Topic 1.4. Binary-coded number systems
Topic 1.5. Algorithms for transferring numbers from one canonical number system to another
Topic 1.6. Algorithms for converting numbers into number systems used in specialized computers and computer systems
Chapter 2. Presentation of numbers in computers and computer systems

Topic 2.1. Presentation of negative numbers in computer arithmetic
Topic 2.2. Overflow of the discharge network and methods of its detection
Topic 2.3. Forms of computer representation of numbers
Chapter 3. Algorithms for performing arithmetic operations
Topic 3.1. Types and algorithms for performing the shift operation
Topic 3.2. Algorithms for adding and subtracting fixed-point numbers in direct and complementary codes
Topic 3.3. Addition-subtraction of binary-decimal operands
Topic 3.4. Algorithms for addition-subtraction of numbers in floating-point form
Topic 3.5. Basic algorithms of computer multiplication in direct code
Topic 3.6. Computer multiplication with rounding
Topic 3.7. Decimal computer multiplication
Topic 3.8. Algorithms of computer multiplication in complementary code
Topic 3.9. Algorithms of accelerated computer multiplication
Topic 3.10. Basic algorithms of computer division
Topic 3.11. Algorithms of accelerated division
Topic 3.12. Other computer division algorithms
Topic 3.13. Algorithms for calculating the square root
Topic 3.14. Tabular methods of calculating elementary functions in KKS
Topic 3.15. Analysis of the accuracy of calculations in KKS
Topic 3.16. Algorithms and errors of computer rounding. Errors in presentation of numbers and arithmetic operations

4. Educational materials and resources

Basic literature

1. *Huth, M., & Ryan, M. (2004). Logic in Computer Science (2nd ed.). Cambridge University Press. Retrieved from* https://www.perlego.com/book/1693624/logic-in-computer-science-modelling-and-reasoning-about-systems-pdf
2. *Rex Page, Ruben Gamboa (2019). Essential Logic for Computer Science. Random House Publishing Group*
3. *John Y. Hsu (2012). Computer Logic. Design Principles and Applications. Springer New York, NY.*

Educational content

5. Methodology of mastering the discipline (educational component)

No. z/p	*Type of training session*	*Description of the training session*
1	*Lecture 1. Numerical systems in computer arithmetic (Part 1)*	*Computer arithmetic. General characteristics of operational resources of computers and computer systems.* *Tasks on self-study: p. 6 No. 1.*
2	*Lecture 2. Numerical systems in computer arithmetic (Part 2)*	*Computer aspects of the theory of calculation systems. Binary-coded number systems.* *Task on self-study: item 6 #2.*

3	*Practical lesson 1*	*Algorithms for transferring numbers from one canonical number system to another.* *Task on self-study: item 6 #3.*
4	*Lecture 3. Numerical systems in computer arithmetic (Part 3)*	*Algorithms for transferring numbers from one canonical number system to another.* *Algorithms for converting numbers into number systems used in specialized computers and computer systems* *Task on self-study: item 6 #4.*
5	*Lecture 5. Representation of numbers in computers and computer systems (Part 1)*	*Presentation of negative numbers in computer arithmetic* *Task on self-study: p. 6 No. 5.*
6	*Practical lesson 2*	*Presentation of negative numbers in computer arithmetic.* *Task on self-study: p. 6 No. 6.*
7	*Lecture 6. Representation of numbers in computers and computer systems (Part 2)*	*Overflow of the discharge network and methods of its detection. Forms of computer representation of numbers* *Tasks on self-study: item 6 #7.*
8	*Lecture 7. Algorithms for performing arithmetic operations (Part 1)*	*Types and algorithms for performing the shift operation.* *Tasks on self-study: item 6 #8.*
9	*Practical lesson 3*	*Overflow of the discharge network and methods of its detection.* *Task on self-study: p. 6 No. 9.*
10	*Lecture 8. Algorithms for performing arithmetic operations (Part 2)*	*Algorithms for adding and subtracting fixed-point numbers in direct and complementary codes.* *Task on self-study: item 6 #10.*
11	*Lecture 9. Algorithms for performing arithmetic operations (Part 3)*	*Addition-subtraction of binary-decimal operands.* *Tasks on self-study: item 6 No. 11.*
12	*Practical lesson 4*	*Algorithms for adding and subtracting fixed-point numbers in direct and complementary codes.* *Task on self-study: item 6 #12.*
13	*Lecture 10. Algorithms for performing arithmetic operations (Part 4)*	*Algorithms for addition-subtraction of numbers in floating-point form.* *Task on self-study: item 6 #13.*
14	*Lecture 11. Algorithms for performing arithmetic operations (Part 5)*	*Basic algorithms of computer multiplication in direct code.* *Task on self-study: item 6 No. 14.*
15	*Practical lesson 5*	*Basic algorithms of computer multiplication in direct code.* *Task on self-study: item 6 No. 15.*
16	*Lecture 12. Algorithms for performing arithmetic operations (Part 6)*	*Computer multiplication with rounding.* *Task on self-study: item 6 No. 16.*

17	*Lecture 13. Algorithms for performing arithmetic operations (Part 7)*	*Decimal computer multiplication.* *Task on self-study: item 6 #17.*
18	*Practical lesson 6*	*Algorithms of computer multiplication in complementary code.* *Task on self-study: item 6 #18.*
19	*Lecture 14. Algorithms for performing arithmetic operations (Part 8)*	*Algorithms of computer multiplication in complementary code.* *Task on self-study: item 6 #19.*
20	*Lecture 15. Algorithms for performing arithmetic operations (Part 9)*	*Algorithms of accelerated computer multiplication.* *Task on self-study: item 6 #20.*
21	*Practical lesson 7*	*Algorithms of accelerated computer multiplication.* *Task on self-study: item 6 #21.*
22	*Lecture 16. Algorithms for performing arithmetic operations (Part 10)*	*Basic algorithms of computer division.* *Task on self-study: item 6 No. 22.*
23	*Lecture 17. Algorithms for performing arithmetic operations (Part 11)*	*Algorithms of accelerated division. Other computer division algorithms.* *Task on self-study: item 6 #23.*
24	*Practical lesson 8*	*Basic algorithms of computer division.* *Task on self-study: item 6 #24.*
25	*Lecture 18. Algorithms for performing arithmetic operations (Part 12)*	*Analysis of the accuracy of calculations in KKS. Algorithms and errors of computer rounding. Errors in presentation of numbers and arithmetic operations.* *Task on self-study: item 6 #25.*
26	*Lecture 19. Algorithms for performing arithmetic operations (Part 13)*	*Analysis of the accuracy of calculations in KKS. Algorithms and errors of computer rounding. Errors in presentation of numbers and arithmetic operations.* *Task on self-study: item 6 No. 26.*
27	*Practical lesson 9*	*Algorithms of accelerated division.* *Task on self-study: item 6 #27.*

6. Independent work of the student

Discipline "Computer logic. Part 2. Computer Arithmetics" is based on independent preparations 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 the lecture 1*	*1*	*1-3*
2	*Preparation for lecture 2*	*1*	*1-3*
3	*Preparation for practical class 1*	*1.5*	*1-3*

4	*Preparation for the lecture 3*	*1*	*1-3*
5	*Preparation for the lecture 4*	*1*	*1-3*
6	*Preparation for practical class 2*	*1.5*	*1-3*
7	*Preparation for the lecture 5*	*1*	*1-3*
8	*Preparation for the lecture 6*	*1*	*1-3*
9	*Preparation for practical class 3*	*1.5*	*1-3*
10	*Preparation for the lecture 7*	*1*	*1-3*
11	*Preparation for the lecture 8*	*1*	*1-3*
12	*Preparation for practical class 4*	*1.5*	*1-3*
13	*Preparation for the lecture 9*	*1*	*1-3*
14	*Preparation for lecture 10*	*1*	*1-3*
15	*Preparation for practical class 5*	*1.5*	*1-3*
16	*Preparation for lecture 11*	*1*	*1-3*
17	*Preparation for lecture 12*	*1*	*1-3*
18	*Preparation for practical class 6*	*1.5*	*1-3*
19	*Preparation for lecture 13*	*1*	*1-3*
20	*Preparation for lecture 14*	*1*	*1-3*
21	*Preparation for practical class 7*	*1.5*	*1-3*
22	*Preparation for lecture 15*	*1*	*1-3*
23	*Preparation for lecture 16*	*1*	*1-3*
24	*Preparation for practical class 8*	*1.5*	*1-3*
25	*Preparation for lecture 17*	*1*	*1-3*
26	*Preparation for lecture 18*	*1*	*1-3*
27	*Preparation for practical class 9*	*1.5*	*1-3*
28	*Preparation for modular control work*	*7*	*1-3*
29	*Preparation for the test*	*8*	*1-3*

Policy and control

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 "Ihor 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 "Ihor Sikorskyi 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

8.1 Current control

The current monitoring of learning results involves students performing practical work. Weighted points of each practical task depend on its complexity and vary in the range from 0 to 3 points.

Practical tasks #1-10 are valued at 1 point.

Practical tasks #11-29 are valued at 2 points.

Practical tasks No. 30-33 are valued at 3 points.

The maximum number of points for practical tasks: 1 point x 10 tasks + 2 points x 19 tasks + 3 points x 4 tasks = 60 points.

The modular test consists of 2 theoretical questions and 1 practical task.

The practical task is valued at 20 points, the theoretical one at 10 points.

Evaluation criteria for a practical question:
20 points – the answer is correct, the calculations are completed in full;
16-18 points - the answer is correct, but not very well supported by calculations;
13-15 points - in general, the answer is correct, but has flaws;
9-12 points – there are minor errors in the answer;
4-8 points – there are significant errors in the answer;
0-3 points - there is no answer or the answer is incorrect.

Evaluation criteria for a theoretical question:
14-15 points – the answer is correct, complete, well-argued;
12-13 points - the answer is correct, detailed, but not very well argued;
9-11 points - in general, the answer is correct, but has flaws;
5-8 points – there are minor errors in the answer;
2-4 points – there are significant errors in the answer;
0 points - there is no answer or the answer is incorrect.

The semester component of the rating scale RS = 100 points, it is defined as the sum of positive points received for answers to practical tasks, evaluations for modular test papers.

The rating scale for the discipline is equal to: R = R p.z. + Rmcr = 60 points + 40 points = 100 points.

Calendar control: is conducted twice a semester as a monitoring of the current state of fulfillment of the syllabus requirements.
At the first assessment (8th week), the student receives "passed" if his current rating is at least 50% of the maximum number of points that the student can receive before the first assessment.
At the second certification (14th week), the student receives "passed" if his current rating is at least 50% of the maximum number of points that the student can receive before the second certification.
With a semester rating (R_c) of at least 60 points and the enrollment of all practical tasks, the student receives credit according to the table (Table of correspondence of rating points to grades on the university scale).

If the student does not agree with the "automatic" grade, he can try to improve his grade by writing a credit test, while his points received for the semester are kept, and the better of the two grades received by the student is assigned ("soft" grading system) .

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

Scores	*Rating*
100-95	Perfectly
94-85	Very good
84-75	Fine
74-65	Satisfactorily
64-60	Enough
Less than 60	Unsatisfactorily
Admission conditions not met	Not allowed

Work program of the discipline (syllabus):

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

Adopted by Computer Systems Software Department (protocol № 12 from 26.04.23)

Approved by the Faculty Board of Methodology (protocol № 10 from 26.05.23)