МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ ІМЕНІ СЕМЕНА КУЗНЕЦЯ

"ЗАТВЕРДЖУЮ"

Проректор з навчально-методичної роботи

_____

Каріна НЕМАШКАЛО

ОСНОВИ АЛГОРИТМІЗАЦІЇ

Робоча програма навчальної дисципліни

Галузь знань	**12 "Інформаційні технології"**
Спеціальність	**121 "Інженерія програмного забезпечення"**
Освітній рівень	**перший (бакалаврський)**
Освітня програма	**Інженерія програмного забезпечення**

Статус дисципліни	**обов'язкова**
Мова викладання, навчання та оцінювання	**англійська**

Завідувач кафедри
інформаційних систем Ірина УШАКОВА

Харків
2021

MINISTRY OF EDUCATION AND SCIENCE OF UKRAINE
SIMON KUZNETS KHARKIV NATIONAL UNIVERSITY OF ECONOMICS

"APPROVE "
Vice-rector for educational and methodical work

Karina NEMASHKALO

FUNDAMENTALS OF ALGORITHMIZATION

Work program of the academic discipline

Knowledge branch	**12 "Information technologies"**
Major	**121 "Software engineering"**
Education level	**First (bachelor)**
Education program	**Software engineering**

Status of the discipline	**compulsory**
Language of teaching, learning and assessment	**English**

Head of
Information systems department Iryna USHAKOVA

Kharkiv
2021

APPROVED
at a meeting of the Information Systems Department
Minutes № 1 of 20.08.2019 p.

Developers:
Gryzun Liudmyla Eduardivna, professor of the Information systems department

Updating and re-approval list
of the work program of the discipline

Academic year	Date of the department meeting	Minute number	Signature of the department head

Introduction

Summary of the academic discipline. Widespread use of information communication technologies and their penetration in all spheres of human activity put forward new, increased requirements for the training of specialists in the branch of information technology. A modern expert in this knowledge domain must have a number of both general scientific and technical competencies. It makes urgent the formation the complex of the pre-service IT-specialists' fundamental knowledge and skills in the area of computer science and software engineering.

In this context, the discipline "Fundamentals of Algorithmization" plays an exclusive part the said specialists' training. It considers such issues as the formalization of the concepts of "algorithm", "complexity of the algorithm" and the study of formal algorithmic systems; general principles of constructing efficient algorithms; modern methods of research and the algorithms analysis; methods and mechanisms for implementing effective algorithms in specific applications; classification of tasks, definition and research of complexity classes; asymptotic analysis of algorithm complexity; research and analysis of recursive algorithms; development of criteria for comparative evaluation of the algorithms quality.

The core aims of the discipline learning are: (1) receiving by the trainees a thorough mathematical training and knowledge of theoretical, methodological and algorithmic fundamentals of information technology for their use in solving applied and scientific problems in the field of information systems and technologies; (2) providing theoretical and engineering training in the design, implementation and use of information systems in business; (3) getting students familiar with state-of-the-art effective algorithms of information processing, as well as the methods of their research and analysis.

Characteristic of the academic discipline

Course	**1**
Semester	**1**
Credits number ECTS	**6**
Form of final assessment	**Exam**

Structural and logical scheme of the discipline learning:

Prerequisites for learning	Postrequisites of learning
Mathematics	Object-oriented programming
Basics of programming	Algorithms and data structures
	Internet programming
	System programming
	Modern technologies of programming

Competences and learning outcomes on the discipline:

Competences	Learning outcomes
SC1.Ability to identify, classify and formulate requirements to the software.	PR12. Apply into practice efficient approaches to the software design.
SC2. Ability to take part in software design including realization of modelling (formalisation), its structure, behavior and function processes.	
SC14. Ability to logical and algorithmical thinking.	
GC1. Ability to abstract thinking, analysis and synthesis	PR13. Know and apply methods of the algorithms development, design of software data structures.
GC2. Ability to use knowledge in practice.	
SC2. Ability to take part in software design including realization of modelling (formalisation), its structure, behavior and function processes.	
SC6. Ability to analyze, sekect and apply methods and means for information security provision (including sybersecurity).	
SC14. Ability to logical and algorithmical thinking.	

Content of the academic discipline

Content module 1. Algorithm concept and its formalization

Theme 1. Algorithm concept. Its main attributes and properties

The purpose and objectives of the discipline, its place in the educational process. The structure of the discipline, recommendations for its study. Organizational and methodological support of the discipline.Algorithm. Intuitive concept of algorithm. Recording methods and properties of algorithms. The need to clarify the concept of algorithm. Requirements for the general algorithmic model. Practical methods of developing the simplest algorithms. The concept of algorithm complexity. Analysis of algorithms for their complexity.

Theme 2. Universal computation models. Post machine.

The concept of a universal computational model. The use of universal computational models to formalize the concept of algorithm. Post machine as a universal computing model. Post machine command system.

Theme 3. Turing machine and machines with unlimited registers

The concept of a universal model "Turing machine". Composition and principle of operation of the Turing machine. Turing machine command system. Capabilities of the Turing machine. The main hypothesis of the theory of algorithms.

Content module 2. Fundamental algorithms of data processing

Theme 4. Methods of algorithms development.

Methods of algorithm development: "divide and conquer", recursion, tree traversal, balancing, dynamic programming, backward programming, "branches and boundaries" method, heuristic and approximate algorithms.

Theme 5. Search and sorting algorithms.

The concept of search in the algorithmic sense. The main types of search and features of their algorithmic implementation. Comparative analysis of the complexity of different search algorithms.

The value of sorting in the implementation of algorithms. Sorting classification. Sorting characteristics. Simple sorting as a way to quickly implement the algorithm. Examples of simple sorting: simple inclusion method, simple exchange method (bubble sorting), shaker sorting, insert sorting, counting sorting, digital sorting. Advantages and disadvantages of simple sorting.

Complex sorting as a way to create efficient algorithms. Examples of complex sorting: Shell sorting, Hoare sorting (quick sorting), merge sorting. Advantages and disadvantages of complex sorting. Comparison of simple and complex sorts.

Theme 6. Algorithms of integers processing.

Euclid's algorithm. The largest common divisor. The least common multiple. Eratosthenes's and Sundaram's sieves. Check for simplicity. Basic theorem of arithmetic. Decomposition of the numbers to prime factors. Euler function. Number of dividers.

Theme 7. Dynamic programming.

The concept of dynamic programming. Basic approaches to solving problems by the method of dynamic programming. Comparative characteristics of dynamic programming and recursion. Matrix calculus. Multiplication of several matrices. Finding the largest common subsequence of sets. Determination of the optimal triangulation of a polygon. Linear programming problems.

Methods of learning and training

Training methods are aimed at activating and stimulating educational and cognitive activities of higher education applicants. During the educational process on the discipline to intensify the educational process it is expected to use a set of methods, such as: problem lectures; work in small groups; presentations; business and role-playing games; case method.

Problem lectures are aimed at developing students' logical thinking. The range of issues of the lecture topic is limited to two or three key points, students' attention is focused on the material that is not widely covered in tutorials, the experience of foreign educational institutions is used to highlight the main conclusions on the issues under consideration. They provide, along with the covering of the main lecture material, the establishment and consideration of a range of problematic issues of debatable nature, which are insufficiently developed in science and are relevant to theory and practice. Lectures of problematic nature are distinguished by in-depth argumentation of the material taught. When teaching lecture material, students are offered questions for independent reflection. The lecturer asks questions that motivate the student to seek a solution to the problem situation. This system forces students to concentrate and start thinking actively in search of the right answer. Problem-based lectures

contribute to the formation of students' independent creative thinking, instill in them cognitive skills. Students become participants in scientific research and problem solving. Problem lectures are implemented within the whole course of the discipline. In particular, they are widely used at learning the **Theme 1. Algorithm concept. Its main attributes and properties; Theme 4. Methods of algorithms development, Theme 5. Search and sorting algorithms; Theme 6. Algorithms of integers processing; Theme 7. Dynamic programming.**

Working in small groups allows to structure lectures or laboratory classes by form and content, creates opportunities for each student to participate in the work on the topic of the lesson, provides the formation of personal qualities and experience of social communication. After covering the problem (when using problem lectures) or summarizing the material (when using mini-lectures), students are invited to unite in groups of 5-6 people and present at the end of the lesson their vision and perception of the material. Within the **Theme 1. Algorithm concept. Its main attributes and properties** students are encouraged to work in pairs (small groups) to work out simple algorithms design and their execution. At learning **Theme 2. Universal computation models. Post machine; Theme 3. Turing machine and machines with unlimited registers** students are involved in the working in small groups to work with Post and Turing machines simulators.

Presentations are included in the educational process on the discipline to present certain achievements, the results of team work, a report on the implementation of individual tasks. One of the positive features of the presentation and its advantages when used in the educational process is the exchange of experience gained by students working in a small group. Especially efficient the presentations preparation is used at learning the **Theme 4. Methods of algorithms development; Theme 5. Search and sorting algorithms; Theme 6. Algorithms of integers processing; Theme 7. Dynamic programming.**

Business and role-playing games is used as a form of activation of students, in which they are involved in the process of staging a certain situation as direct participants in events. For example, during a laboratory lesson, students can be divided into groups, each of which receives from the teacher a specific task, the implementation of which must be performed using different approaches. Role-playing games are implemented at learning the **Theme 5. Search and sorting algorithms**, which enables students to realize deeper the theoretical essence and practical significance of the basic algorithms. In addition, the roleplaying provides involving students into process of the active kinetic imitation of the algorithms.

Case method is used during the classes as a method of analysis of specific situations, which allows to bring the learning process closer to the real practical activities of specialists and involves consideration of production, management and other situations, complex conflicts, problem situations, incidents in the process of studying the material. The method is relevant to apply at learning **Theme 1. Algorithm concept. Its main attributes and properties** especially for understanding of the concept of algorithm complexity and its analysis.

Assessment system of learning outcomes

The system of assessment of formed students' competencies takes into account the types of classes, which according to the curriculum of the discipline include lectures, laboratory classes, as well as independent work. Assessment of the formed competencies of students is carried out according to the accumulative 100-point system. control measures include:

current control, which is carried out for the semester during lectures, laboratory classes and is estimated by the amount of points scored (maximum amount is 60 points; the minimum amount that allows a student to take the exam is 35 points);

final / semester control, which is conducted in the form of a semester exam, according to the schedule of the educational process (maximum amount is 40 points; the minimum amount that allows a student to pass the exam is 25 points).

The procedure for conducting current assessment of students' knowledge. Ongoing monitoring includes student assessment during:

Lectures - active work during the lecture (1 point for each lecture) on condition the student gives the correct answers to the control questions during the survey, participates in the discussion of the results of solving the problem situation posed at the beginning of the lesson. The total number of points is 8 points;

Laboratory works which aim to check the level of readiness of the student to perform a particular work and solve problems. Assessment involves the defense of reports on laboratory works (4-5 points for each work), provided that the student's level of knowledge meets the criteria. The total number of points is 48 points. Assessment of students' knowledge during laboratory classes and performance of individual tasks is carried out according to the following criteria: understanding, degree of mastering the theory and methodology of the problems under consideration; the degree of mastering the discipline content; acquaintance with the recommended literature, and also with the contemporary literature on the considered issues; ability to combine theory with practice at solving problems, making calculations in the process of performing individual tasks and tasks submitted for consideration in the classroom; logic, structure, style of presentation of the material in written works and in presentations in front of the audience, the ability to justify their position, to generalize information and draw conclusions; arithmetic correctness of performance of individual and complex calculation task; ability to conduct critical and independent assessment of certain problematic issues; the ability to explain alternative views and having their own point of view, position on a particular issue; application of analytical approaches; quality and clarity of reasoning; logic, structuring and validity of conclusions on a specific problem; independence of work performance; correctness of material submission; use of methods of comparison, generalization of concepts and phenomena; representation of the work;

Extracurricular independent work which is directed on the preparation to problem lectures and solving problems of laboratory classes. The results of students' extracurricular independent work is assessed during their revealing at lectures and laboratory classes. The general criteria for assessing the results of students' extracurricular independent work are: depth and strength of knowledge, level of thinking, ability to systematize knowledge on individual topics, ability to draw sound conclusions, mastery of categorical apparatus, skills and techniques of practical tasks, ability to find necessary information, carry out its systematization and processing, self-realization at laboratory classes;

Test paper that covers the curriculum of the discipline and expects the determination of the level of knowledge and the degree of mastery of competencies by students. Test paper consists of 4 tasks which expect the solution of typical tasks of the course and allows to diagnose the level of theoretical training of the student and his level of competence in the discipline. The total number of points is 4 points.

Final control of knowledge and competencies of students in the discipline is carried out based on a semester exam, the aim of which is to test students' understanding of the program material in general,

logic and relationships between individual sections, ability to creatively use accumulated knowledge, ability to formulate their attitude to a particular problem of the discipline etc.

The result of the semester exam is evaluated in points (maximum number - 40 points, the minimum number of credits - 25 points) and is affixed in the appropriate column of the examination "Information of success".

The final grade in the discipline is calculated taking into account the points obtained during the exam and the points obtained during the current control of the accumulative system. The total result in points for the semester is: "60 or more points - credited", "59 or less points - not credited".

The final score is set according to the scale given in table below.

Assessment scale: national and ECTS

Total score on a 100-point scale	ECTS assessment scale	Assessment on the national scale	
		for exam, differentiated test, course project (work), practice, training	for pass
90 – 100	A	excellent	pass
82 – 89	B	good	
74 – 81	C		
64 – 73	D	satisfactory	
60 – 63	E		
35 – 59	FX	unsatisfactory	not pass
1 – 34	F		

Rating-plan of the academic discipline

Theme	**Forms and kinds of learning**		**Assessment forms**	**Max grade**
Theme 1. Algorithm concept. Its main attributes and properties	***Classroom work***			
	Lecture	Lecture on the issues: The purpose and objectives of the discipline, its place in the educational process. The structure of the discipline, recommendations for its study. Organizational and methodological support of the discipline		
	Lecture	Lecture on the issues: Algorithm. Intuitive concept of algorithm. Recording methods and properties of algorithms (Problem lecture**).**	active work during the lecture	
	Lecture	Lecture on the issues: The need to clarify the concept of algorithm. Requirements for the general algorithmic model (Case method).	active work during the lecture	

Theme	Forms and kinds of learning		Assessment forms	Max grade
	Lecture	Lecture on the issues: The concept of algorithm complexity. Analysis of algorithms for their complexity (Problem lecture).	active work during lecture	
	Laboratory class	Laboratory work № 1. A flowchart as a way of algorithms expression (Working in small groups).	defense of the report on laboratory work	5
	Laboratory class	Laboratory work № 2. Practical methods of developing the simplest algorithms	defense of the report on laboratory work	5
	Independent work			
	Tasks for the independent elaboration	Lecture material learning. Preparing for the laboratory classes.		
Theme 2. Universal computation models. Post machine.	***Classroom work***			
	Lecture	Lecture on the issues: The concept of a universal computational model. The use of universal computational models to formalize the concept of algorithm.		
	Laboratory class	Laboratory work № 3. Post machine as a universal computing model. Post machine command system (Working in small groups).	defense of the report on laboratory work	5
	Independent work			
	Tasks for the independent elaboration	Lecture material learning. Preparing for the laboratory classes. Work in Post machine simulator.	express-survey on the topic of the lab.work	
Theme 3. Turing machine and machines with unlimited registers	***Classroom work***			
	Lecture	Lecture on the issues: The concept of a universal model "Turing machine". Composition and principle of operation of the Turing machine. The main hypothesis of the theory of algorithms.		
	Laboratory class	Laboratory work № 4. Turing machine command system. Capabilities of the Turing machine (Working in small groups).	defense of the report on laboratory work	5
	Independent work			
	Tasks for the independent elaboration	Lecture material learning. Preparing for the laboratory classes. Work in Turing machine simulator.	express-survey on the topic of the lab.work	

Theme	Forms and kinds of learning		Assessment forms	Max grade
Theme 4. Methods of algorithms development.	***Classroom work***			
	Lecture	Lecture on the issues: Methods of algorithm development. Common characteristics.		
	Lecture	Lecture on the issues: Methods of algorithm development. "Divide and conquer" method as a common method for search and sorting problems (Problem lecture).	active work during lecture	
	Laboratory class	Laboratory work № 5. Solving algorithmic problems based on "Divide and conquer" method of algotithmic technique.	Presentations	5
	Independent work			
	Tasks for the independent elaboration	Lecture material learning. Preparing for the laboratory classes. Learning recommended and additional literature.	express-survey on the topic of the lab.work	
Theme 5. Search and sorting algorithms	***Classroom work***			
	Lecture	Lecture on the issues: The concept of search in the algorithmic sense. The main types of search and features of their algorithmic implementation. Comparative analysis of the complexity of different search algorithms (Business and role-playing games**)**.	active work during lecture	
	Laboratory class	Laboratory work № 6. Simple search and binary search in arrays. Comparative analysis of the complexity.	Presentations	5
	Lecture	Lecture on the issues: The value of sorting in the implementation of algorithms. Sorting classification. Sorting characteristics. Simple sorting as a way to quickly implement the algorithm. Examples of simple sorting. Advantages and disadvantages of simple sorting**.**		
	Laboratory class	Laboratory work № 7. Simple sorting in arrays.	defense of the report on laboratory work	5
	Lecture	Lecture on the issues: Complex sorting as a way to create efficient algorithms. Examples of complex sorting: Shell sorting, Hoare sorting (quick sorting), merge sorting. Advantages and disadvantages of complex sorting. Comparison of simple and complex sorts (Problem lecture).	active work during lecture	
	Laboratory class	Laboratory work № 8. Complex sorting in arrays on the example of merge sorting (Business and role-playing games).	defense of the report on laboratory work	5
	Independent work			
	Tasks for the independent elaboration	Lecture material learning. Preparing for the laboratory classes. Learning recommended and additional literature.	express-survey on the topic of the lab. work	

Theme	Forms and kinds of learning		Assessment forms	Max grade
Theme 6. Algorithms of integers processing	***Classroom work***			
	Lecture	Lecture on the issues: Euclid's algorithm. The largest common divisor. The least common multiple. Eratosthenes's and Sundaram's sieves. Check for simplicity. Basic theorem of arithmetic. Decomposition of the numbers to prime factors. Euler function. Number of dividers (Problem lecture).	active work during lecture	
	Laboratory class	Laboratory work № 9. Euclid's algorithm realization.	defense of the report on laboratory work	
	Laboratory class	Laboratory work № 10. Prime numbers problems	Presentations	5
	Independent work			
	Tasks for the independent elaboration	Lecture material learning. Preparing for the laboratory classes. Learning recommended and additional literature.	express-survey on the topic of the lab. work	
Theme 7. Dynamic programming	***Classroom work***			
	Lecture	Lecture on the issues: The concept of dynamic programming. Basic approaches to solving problems by the method of dynamic programming. Comparative characteristics of dynamic programming and recursion. Matrix calculus. Finding the largest common subsequence of sets. Linear programming problems (Problem lecture).	active work during lecture	
	Laboratory class	Laboratory work № 11. Basic approaches to solving problems by the method of dynamic programming	defense of the report on laboratory work	5
	Laboratory class	Laboratory work № 12. Comparative characteristics of dynamic programming and recursion	Presentations	5
	Laboratory class	Test paper		5
	Independent work			
	Tasks for the independent elaboration	Lecture material learning. Preparing for the laboratory classes and test paper. Learning recommended and additional literature.	express-survey on the topic of the lab. work	
		For the term		60
		Exam		40
		In total		100

Recommended literature

Main literature

1.Algorithms Fundamentals. E-resourse. - available at: https://brilliant.org/courses/computer-science-algorithms/

2.Bhargava A. Y Grokking Algorithms - An illustrated guide for programmers and other curious people, 2018**,** available at:

https://github.com/RbkGh/Free-Algorithm-Books/blob/master/book/Grokking%20Algorithms%20-%20An%20illustrated%20guide%20for%20programmers%20and%20other%20curious%20people.pdf

3. Jay Introduction to Algorithms for Beginners and Aspiring Programmers. Tutorials, 2019, available at: https://www.thecodingdelight.com/introduction-algorithms/

Additional literature

4. Bilousova L., Gryzun L., Zhytienova N., Pikalova V. (2019) Search algorithms learning based on cognitive visualization. ICT in Education, Research, and Industrial Applications: Integration, Harmonization, and Knowledge Transfer. Conference proceedings (2387). P. 472-478. ISSN 1613-0073. URL: http://ceur-ws.org/Vol-2387/20190472.pdf

5. Матвієнко М.П. Алгоритми та структури даних: навчальний посібник. / М. П. Матвієнко. – Київ: Видавництво Ліра-К, 2018. — 340 с.

6.Knuth, D E The Art of Computer Programming (TAOCP). Vol. 1.Retrieved May 20, 2019, available at: https://cs.stanford.edu/~knuth/taocp.html

7. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest Clifford Stein Introduction to Algorithms - Third Edition, 2019, available at: https://github.com/RbkGh/Free-Algorithm-Books/blob/master/book/Introduction%20to%20Algorithms%20-%20Third%20Edition.pdf

Internet resources

8. Algorithms and data structures [E-resource]. – Available at : https://www.lektorium.tv/course/22823?id=22823.

9. Algorithms complexity [E-resource]. – Available at : http://habrahabr.ru/post/188010.

10. Algorithms and Data Structures [E-resource]. – Available at: https://sites.google.com/site/indy256/algo_cpp.

11. Fundamentals of algorithmization (121, 122) [PNS-course]. – Available at : https://pns.hneu.edu.ua/course/view.php?id=5572

12. Sorting Algorithm Animations [E-resource]. – Available at : http://www.sorting-algorithms.com.