

SYLLABUS

INTERNATIONAL EUROPEAN
UNIVERSITY



EUROPEAN SCHOOL
OF BUSINESS



SYLLABUS



Discipline

 Algorithms and Data structures

Lecturer (s)

 Oleksandr Falovskyi, Associate Professor at the Department of Information Technology, PhD in Technology

Discipline page

 <https://it.ieu.edu.ua/pro-yeash/struktura-yeash/kafedra-informatsiinykh-tekhnologii/sklad#zzz-004>

Consultations

Online:  Friday 4 p.m. – 6 p.m.

Offline  -

Contact number

 0966562963

E-mail

 oleksandr_falovskyi@ieu.edu.ua

Discipline page

 <https://dist.ieu.edu.ua/course/view.php?id=454>

Form of final control

Test

Fail test

Exam



SYLLABUS



1 Brief discipline annotation

The Algorithms and Data structures academic discipline is designed for Bachelors of the knowledge area: 12 Information Technology, specialty: 121 Software Engineering. It is one of the fundamental natural science disciplines for future software developers.

2 Background for studying discipline

The discipline program is based on complete general secondary education or Junior specialist's degree.

3 Goal and objectives of the discipline

The goal of the discipline is to provide students with the ability to efficiently build group work and communication in a professional context in software engineering; to work fruitfully with colleagues, use the concepts of group dynamics, listening strategies, conflict resolution; to persuade and negotiate, use the principles of effective oral communication; to create presentations of appropriate quality.

Key objectives of the discipline:

- to provide students with knowledge about the role of data algorithms and structures in software development;
- to introduce different types of data structures, their differences and suitability for certain classes of tasks;
- to reveal the essence of the methodology of designing and building algorithms;
- to develop students' understanding of the basic approaches to constructing the main classes of algorithms;
- to provide students with knowledge, skills and abilities to independently build algorithms for different classes of tasks and independently choose the necessary data structures;
- to develop students' skills in testing algorithms at the design stage;
- to be able to apply the acquired knowledge

4 Learning outcomes

After learning the discipline, students should

know:

- data types and structures;
- theoretical foundations of building algorithms;
- types of data processing tasks;
- main characteristics of algorithms by their classes and correlation with types of data processing tasks;
- ways to model the process of algorithm execution;

be able to:

- use acceptable data types depending on the types of data processing tasks;
- apply the acquired knowledge in practical programming;
- independently create algorithms for solving practical problems;
- create descriptions of algorithms as a data processing model;
- independently check the correctness of the built algorithms;

5 Credits

4 credits

6 Discipline structure

Sections and topics	Type of classes/hours		
	Lectures	Practical classes	Independent work
Content section 1. General concepts			
Topic 1.1. General overview of data processing tasks as a set of algorithms over data structures	0.5		
Topic 1.2. Types of data. Addressing.	0.5	1	
Topic 1.3. Algorithms. Their structure. Steps of construction. Interaction between data types and algorithms	0.5	1	
Content section 2. Sorting algorithms			
Topic 2.1. General formulation of the sorting task. Data structures used in sorting tasks.	1	1.5	
Topic 2.2. Types of sorting algorithms. Methods of organizing the search in sorting algorithms	1	2	
Content section 3. Search algorithms			
Topic 3.1. Search tasks reduced to the search in depth class. The mechanism of the backtracking method.	1	1.5	
Topic 3.2. Search tasks reduced to the search in width class. The mechanism of the branches and borders method.	0.5	1.5	
Topic 3.3. Data structures used in search tasks. Trees. Heaps.	1	2	
Topic 3.4. Data structures used in search tasks. Graphs.	0.5	1.5	
Topic 3.5. Search problems that are supercomplex to solve. P- and NP-complete search problems	0.5	0.5	
Content section 4. Hash functions			
Topic 4.1. Hash functions as a special type of addressing in data structures	0.5	1	
Topic 4.2. Hash functions and blockchain technology	0.5	1	
Total	8	16	



SYLLABUS



7

List of obligatory tasks

1. To master the basic types of data and their inherent methods of addressing, to know the classes of tasks where these types of data should be used
2. To understand the organization of linear data types and classes of tasks where they are appropriate to use
3. To understand the organization of non-linear data types and classes of tasks where they are appropriate to use
4. To understand the types of addressing in data structures
5. To know the basic principles and steps of building algorithms
6. To learn the basic approaches to building sorting algorithms
7. To learn the basic approaches to building search algorithms
8. To learn the basic approaches to building algorithms for modifying information using different types of data

8

List of selective tasks

1. To describe the methodology for identifying the types and structures of data that will be needed to implement the algorithm
2. To compare the main methods of information processing in algorithms
3. To describe the principles of constructing algorithms based on learning
4. Data structures used in the construction of databases.
5. Features of designing algorithms for a class of supercomputable (NP-complete) problems

9

Discipline features

Period of teaching	Semester	International discipline integration	Year of study	Courses: general training/professional training/elective
1 semester	4	Available	1	professional training

10

Hardware and software

Personal computer, office software packages, environment for building graphical information schemes



SYLLABUS



11 Assessment system and requirements

As part of discipline teaching, one carries out the current and final control of students' knowledge. The final grade is given according to the total rating of students.

The results of the current control of students' knowledge is assessed in general between 0 and 60 points. Students are admitted to the final control if they fulfil the requirements of the training program and obtain at least 36 points for the current learning activity.

Final assessment of students' knowledge is conducted in the form of exam.

The maximum amount of scores that can be obtained during the exam is 40 points.

The overall points of the discipline are 100. The total grade for the discipline is given according to the national and European scale.

12 Absence policy

Teaching of the discipline is based on cutting-edge educational technologies aimed at increasing the level of students' interest in the course, providing theoretical and practical knowledge of the discipline.

To activate the learning and cognitive activity of students, the discipline includes the consolidation of knowledge obtained at the lecture and acquisition of practical skills in lecture topics during laboratory classes.

13 Absence policy

Points are not given for missed lectures. If students miss a laboratory work, they should perform the homework before the next laboratory work and present the results to the lecturer.

Students who have missed classes without valid reasons and have not participated in current control activities are not admitted to the final semester control. In this case, a mark 'non-admission' is put in the exam record on the day of the exam. Repeated taking of the exam of the discipline is appointed in case of accomplishing all types of educational, independent (individual) work stipulated by the working program of the academic discipline and is carried out according to the approved schedule of academic failure liquidation.

14 Policy of late task performance

Tasks submitted later are assessed with a lower grade.

15 Academic integrity policy

Participants in the educational process rely on the academic integrity principles. One should provide references to sources of information when using someone else's ideas, statements, data, as well as verified information.



SYLLABUS



16

Recommended sources of information

Primar

1. Wirth N. Data algorithms and structures: translated from English / N. Wirth. – Moscow: Mir, 2014. – 360 p.
2. Aho A., Hopcroft D., Ullman D. Data structures and algorithms: Translated from English. – M.: Williams Publishing House, 2019. – 400 p.
3. Knuth D. The art of programming. Basic algorithms. Volume 1. – M.: Williams Publishing House, 2002. – 720 p.
4. Knuth D. The Art of Programming. Semi-numerical Algorithms. Volume 2. – M.: Williams Publishing House, 2001. – 832 p.
5. Knuth D. The Art of Programming. Combinatorial algorithms. Volume 4A. – M.: Williams Publishing House, 2013. – 960 p.
6. Krenevych A.P. Data algorithms and structures. Textbook. – K.: Kyiv University Publishing Center, 2021. – 200 p.
7. Matviienko M.P. Data algorithms and structures: study guide / M.P. Matviienko. – Kyiv: Lira-K Publishing House, 2014. – 340 p.

Additional:

8. Ageev M.I., Alik V.P., Galis R.M., Markov Y.I. Library of algorithms 1b-50b (Reference manual). – M.: Soviet Radio, 1975. – 176 p. (P. 43).
9. Borodkina I.L., Borodkin H.O. Theory of algorithms: Guide for students of higher education institutions. - K.: National University of Life and Environmental Sciences of Ukraine, 2018. – 231 p.
10. Dasgupta S., Papadimitriou H., Vazirani U. Algorithms. – M.: ICNMO, 2014. – 320 p.
11. Shen A. Programming: theorems and problems. – 3rd ed. – M.: ICNMO, 2007 – 296 p.
12. Cryptography from papyrus to computer. – <http://www.taina.dinamikweb.ru/a/index.htm> (<http://www.taina.dinamikweb.ru/a/index-prosteshie-algoritmy-generacii-chast-2.htm>)
13. Lviv M.S., Spivakovskiy O.V. Fundamentals of algorithmization and programming: Study guide. – Kherson, 1997. – 371 p.
14. Tkachuk V.M. Data algorithms and structures: Study guide. – Ivano-Frankivsk: Vasyl Stefanyk Precarpathian National University Publishing House, 2016. – 286 p. (pp. 227-233).
15. Hughes J., Mitchum J. Structural Approach to Programming – Moscow: Mir, 1980. – 280 p.

Internet resources

1. E-Olymp [Electronic resource] – Available at: [www.e-olymp.com/courses.prometheus.org.ua/courses/Prometheus/CS50/2016_T1/about](http://www.e-olymp.com/courses/prometheus.org.ua/courses/Prometheus/CS50/2016_T1/about)
2. Sorted algorithm animation [Electronic resource] – Available at: <https://www.toptal.com/developers/sorting-algorithms>
3. Problem Solving with Algorithms and Data Structures using Python [Electronic resource] – Available at: <https://runestone.academy/runestone/books/published/pythonds/index.html>.

17

Tips on successful study during the course

Note: examine lecture materials and perform laboratory tasks synchronously with the curriculum. Thus, your abilities and insistence will be the key to success!