

ADVANCED JAVA PROGRAMMING COURSE

COURSE SYLLABUS

Code and name of specialty	121 Software Engineering	Institute / faculty	Faculty of Computer Science and Software Engineering
	122-Computer Science		
	126-Information Systems and technologies		
Program name	Software Engineering	Department	Software Engineering and Management Information Technologies
	Computer Science and Intelligent Systems		
	Information Systems Software		
Type of program	Educational and Professional	Language of instruction	Ukrainian, English

LECTURER

Full name, e-mail

**DVUKHHLAVOV Dmytro,
Dmytro.Dvukhhlavov@khpi.edu.ua**



Ph.D., Associate Professor, Associate Professor of Software Engineering and Information Technology Management. Has prepared and published more than 40 publications, 1 article in publications indexed in Scopus, 2 textbooks, 2 guidelines for making practice tasks.
h-index = 3, i10-index = 0 y Google Academy-[https://scholar.google.com/citations?user= OAzyFg8AAAAJ&hl=ru](https://scholar.google.com/citations?user=OAzyFg8AAAAJ&hl=ru); identifier ORCID-[https://orcid.org/ 0000-0002-3361-3212](https://orcid.org/0000-0002-3361-3212)).
Leading lecturer of the courses: Advanced Java programming course (*Bachelors*) (*Ukrainian*), Java-based web applications (*Bachelors*) (*Ukrainian*), Architecture and Design of Software (part 1) (*Bachelors*) (*English and Ukrainian*), Architecture and Design of Software (part 2) (*Bachelors*) (*English and Ukrainian*).

GENERAL DESCRIPTION OF THE COURSE

Summary

The discipline «Advanced Java programming course» is study discipline from the cycle “Optional student disciplines of the profile training” of preparation level "bachelor" on specialty 121 “Software Engineering”. It is taught in the fourth semester in the amount of 1890 hours (6 ECTS credits), in particular: lectures - 32 hours, laboratory classes - 32 hours, independent work - 116 hours. Final control – credit.
The presentation of the discipline provides an increase in the level of readiness of the Java developer to create software for efficient data processing in corporate information systems.

Course objectives	The course objective is to deepen the system of knowledge about the capabilities of Java components and technologies that can be used to work with data sets of different types and purposes (JCF, Stream API, file system), for efficient processing of text data and time data, for software creation of text and tabular documents, as well as the organization of multi-user coordinated access to data, as well as practice skills of practical application of acquired knowledge by solving applications that are typical in data processing in enterprise-level information systems						
Types of classes and control	Lectures, laboratory classes, consultations. Final control – credit.						
Terms	4						
Student workload (credits) / Type of course	4 / Selective	Lectures (hours)	32	Workshops (hours)	32	Independent work (hours)	116
Program competencies	<p>121-GC 2. Ability to apply knowledge in practical situations.</p> <p>121-PC20. Ability to apply fundamental and interdisciplinary knowledge to successfully solve software engineering problems.</p> <p>121-PC26. Ability to algorithmic and logical thinking.</p> <p>122-GC1. Ability to abstract thinking, analysis and synthesis.</p> <p>122-GC2. Ability to apply knowledge in practical situations.</p> <p>122-GC6. Ability to learn and master modern knowledge.</p> <p>122-GC7. Ability to search, process and analyze information from various sources.</p> <p>122-GC8. Ability to generate new ideas (creativity).</p> <p>122-PC8. Ability to design and develop software using different programming paradigms: generalized, object-oriented, functional, logical, with appropriate models, methods and algorithms of calculations, data structures and management mechanisms.</p> <p>122-PC10. Ability to apply methodologies, technologies, and tools to manage the life cycle processes of information and software systems, information technology products and services according to customer requirements.</p> <p>122-PC12. Ability to ensure the organization of computational processes in information systems of various purposes, taking into account the architecture, configuration, performance indicators of operating systems and system software.</p> <p>126-GC 1. Ability to abstract thinking, analysis and synthesis.</p> <p>126-GC 2. Ability to apply knowledge in practical situations.</p> <p>126-GC 3. Ability to understand the subject area and professional activity.</p> <p>126-GC 5. Ability to learn and master modern knowledge.</p> <p>126-PC 1. Ability to analyze the object of design or operation and its subject area.</p> <p>126-PC 4. Ability to design, develop and use tools for the implementation of information systems, technologies and infocommunications (methodological, informational, algorithmic, technical, software and others).</p>						

Learning outcomes	<p>121-PO13. Know and apply methods of algorithm development, software design and data and knowledge structures.</p> <p>121-PO23. Be able to document and present the results of software development.</p> <p>122-PLO5. Design, develop and analyze algorithms for solving computational and logical problems, evaluate the efficiency and complexity of algorithms based on the use of formal models of algorithms and computational functions.</p> <p>122-PLO9. Develop software models of subject areas, choose a programming paradigm from the standpoint of convenience and quality of its application to implement methods and algorithms that solve problems in the computer science field.</p> <p>122-PLO17. Apply for the construction of intelligent management systems theoretical and practical foundations of modern management theory, design intelligent systems using modern information processing technologies and methods of computational intelligence.</p> <p>126-PLO 3. To use basic knowledge of informatics and modern information systems and technologies, programming skills, technologies of safe work in computer networks, methods of creation of databases and Internet resources, technologies of development of algorithms and computer programs in high-level languages with application of project-oriented programming to solve problems of design and use of information systems and technologies.</p> <p>126-PLO 4. Conduct a systematic analysis of design objects and justify the choice of structure, algorithms and methods of information transfer in information systems and technologies.</p> <p>126-PLO 6. Demonstrate knowledge of the current level of information systems technology, practical skills of programming and use of applied and specialized computer systems and environments for their implementation in professional activities.</p> <p>126-PLO 7. Justify the choice of technical structure and develop appropriate software that is part of information systems and technologies.</p>
Teaching and learning methods	<p>The main method of teaching during lectures is the explanatory-illustrative method. To intensify cognitive activity, students' speeches and organization of discussions on certain issues of lectures are provided.</p> <p>Execution of laboratory works involves the creation of student code to solve several problems on the topic. There are no ready-made algorithms for solving, which encourages the manifestation of creative activity of students. At least one individual task is formulated in the tasks for each laboratory work.</p>
Forms of assessment (continuous assessment CAS, final assessment FAS)	<p>Assimilation of the theory is tested in the form of a rapid survey during lectures (CAS), a survey or automated testing at the beginning of laboratory work (CAS). Control of mastering the material for self-study involves the preparation and defense of abstracts on individual topics (2 abstracts) (CAS). The level of practical skills is tested in laboratory work, which is mainly performed on an individual basis (CAS). Final / semester control is carried out in the form of a test, which involves the development of a web application for an individual task in a limited time (FAS)</p>

ASSESSMENT AND GRADING

Ranges of points corresponding to grades	Total score (points) for all types of learning activities	ECTS grading scale	The national grading scale	Allocation of grade points	Assimilation of theory (topics of independent work)	30 points	
	90-100	A	excellent		good	Working out the tasks of the laboratory workshop	50 points
	82-89	B					
	74-81	C	satisfactory		Passing credit	20 points	
	64-73	D					
	60-63	E					
	35-59	FX	Unsatisfactory (with the exam retake option)		Summary	100 points	
0-34	F	Unsatisfactory (with mandatory repetition of the course)					

Course policy

Students are required to attend classes according to the schedule. In the absence of a student at the lecture, he works out a syllabus of lectures before the next lesson. Participation in laboratory work involves the need to repeat the lecture material and self-study of recommended sources. At the beginning of the laboratory there is an experience of students for the materials of lectures and independent work. Performing laboratory tasks requires prior preparation and advance processing of all necessary materials for productive discussions during the lesson and their operational implementation. All laboratory work is required to obtain a final grade in the discipline. An important element of training is the need to adhere to the schedule of presentation of laboratory results and abstracts. For delay in execution without an officially confirmed reason, the score is reduced.

COURSE STRUCTURE AND CONTENT

Topic	Content	Laboratory class	Independent work
Topic 1	Java Collections Framework. Hierarchy of classes. Features of creating and using classes. (4 hours)	Laboratory class 1	Research of properties of construction of Java collections and principles of their using (6 hours) (PT70)
Topic 2	Processing of terms using regular expressions in Java (4 hours)	Laboratory class 2	Research of principles of work with terms with use of regular expressions (4 hours) (PT71)
			Classification of collections: hierarchy, methods of classes, features of construction and use. Language structure for constructing regular expressions. Means of creating regular expressions.

Topic 3	Internationalization, localization, work with dates and time in Java (4 hours)	Laboratory class 3	Research of Java 8 technologies for processing dates, time and localization of applications (4 hours) (PT72)	Classification of locales and languages. Classes and methods of the java.time library.
Topic 4	Java 8 technologies for working with the file system (4 hours)	Laboratory class 4	Research of Java 8 technologies for work with file system (2 hours) (PT75)	Classes and methods of the java.nio and java.nio2 libraries
Topic 5	Processing of text and tabular documents in Java (4 hours)	Laboratory class 5	Research of principles of creation of text documents and tables by Java methods (4 hours) (PT76)	Object structure of a text document (based on Word) and a table (based on Excel). Known technologies for working with text and spreadsheet documents in Java.
Topic 6	Generalized and functional programming in Java. Parameterized classes and methods. Lambda expressions. Functional interfaces (4 hours)	Laboratory class 6	Research of principles of realization of the generalized and functional programming on Java (4 hours) (PT73)	Typical tasks solved with the use of lambda expressions and functional interfaces. Techniques for using lambda expressions and functional interfaces
Topic 7	Purpose and features of using the Stream API (4 hours)	Laboratory class 7	Research of the principles of application of Stream API methods for data set processing (4 hours) (PT74)	Typical tasks solved using the Stream API
Topic 8	Implementation of multithreading in Java (4 hours)	Laboratory class 8	Research of principles of realization of multithreading used methods of Java (4 hours) (PT77)	Typical tasks, the implementation of which requires multithreading. Working with a thread pool.

RECOMMENDED READING

Compulsory

- 1 Bloch, J. (2017). Effective Java: 3rd Edition. Addison Wesley, 412 p.
- 2 Schildt, H. (2018). Java: A Beginner's Guide: 8th Edition, McGraw-Hill Education, 684 p.
- 3 Schildt, H. (2018). Java: The Complete Reference: 11th Edition. McGraw-Hill Education, 1208 p.
- 4 Horstmann, C. S. (2018). Core Java. Volume I. Fundamentals: 11th Edition. Prentice Hall, 889 p.
- 5 Horstmann, C. S. (2019). Core Java. Volume II. Advanced Features (Core Series): 11th Edition. Pearson, 1040 p.
- 6 Kishori Sharan. (2014). Beginning Java 8 Fundamentals. Language Syntax. Arrays. Data Types. Objects and Regular Expressions. Apress, 828.
- 7 Java 8 Programming: Black Book by DT Editorial Services. Dreamtech Press, 2015. 1052 p.
- 8 Urma Raoul-Gabriel, Fusco Mario, Mycroft Alan. (2015). Java 8 in Action: Lambdas, streams, and functional-style programming. Manning Publications Co, 497 p.
- 9 Apache POI – the Java API for Microsoft Documents. [Electronic resource]. Access mode: <https://poi.apache.org/>

Recommended

INFORMATION RESOURCES IN INTERNET

- 1 The Java Tutorials [Electronic resource]. Access mode: <https://docs.oracle.com/javase/tutorial/tutorialLearningPaths.html>
- 2 <https://metanit.com/java/>
- 3 <https://www.javaguides.net/>
- 4 <https://www.javatpoint.com/>
- 5 <https://betacode.net/>

ACADEMIC INTEGRITY

The student must adhere to the Code of Ethics of Academic Relations and Integrity of NTU "KhPI": to show discipline, politeness, friendliness, honesty, responsibility

The content of this syllabus is consistent with the course program.