Syllabus

Course Program

Java Data Science solutions

Specialty
121 – Software Engineering
122 – Computer Science

Institute
Institute of Computer Science and Information Technology

Educational program
Software Engineering
Computer Science and Intelligent Systems

Department
Software Engineering and Management Intelligent Technologies (321)

Level of education
Bachelor's level

Course type
Special (professional), Elective

Semester
7

Language of instruction
English, Ukrainian

Lecturers and course developers

Dmytro Dvukhhlavov

dmytro.dvukhhlavov@khpi.edu.ua
Ph.D., Associate Professor, Associate Professor of Software Engineering and Information Technology Management.

Google Scholar: https://scholar.google.com/citations?user=0AzyFg8AAAAJ&hl
ORCID: https://orcid.org/0000-0002-3361-3212
Scopus: https://www2.scopus.com/authid/detail.uri?authorId=57211294555
Web of Science: https://www.webofscience.com/wos/author/record/E-8279-2019).
More about the lecturer on the department's website

General information

Summary

Studying the discipline provides the future Java-developer with a set of knowledge and skills necessary for solving Data Science tasks in Java, in particular tasks of processing and visualizing data arrays, as well as machine learning.

Course objectives and goals

The purpose of teaching the discipline is to provide the student with knowledge about the capabilities of Java and means of implementing some Data Science tasks in this language, as well as practicing the skills of their use on examples of real tasks in the field of control automation.

Format of classes

Lectures, laboratory classes, consultations, self-study. Final control in the form of a credit.

Competencies

121 - Software Engineering
GC 2. Ability to apply knowledge in practical situations.
PC15. Ability to develop architectures, modules and components of software systems.

PC19. Knowledge of information data models, the ability to create software for data storage, retrieval and processing.
PC20. Ability to apply fundamental and interdisciplinary knowledge to successfully solve software engineering problems.
PC26. Ability to algorithmic and logical thinking.

122 - Computer Science and Intelligent Systems
GC1. Ability to abstract thinking, analysis and synthesis.
GC2. Ability to apply knowledge in practical situations.
GC6. Ability to learn and master modern knowledge.
GC7. Ability to search, process and analyze information from various sources.
GC8. Ability to generate new ideas (creativity).
PC3. Ability to think logically, build logical conclusions, use formal languages and models of algorithmic calculations, design, develop and analyze algorithms, evaluate their efficiency and complexity, solvability and insolvability of algorithmic problems for adequate modelling of subject areas and creation of software and information systems.
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.
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.
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.

Learning outcomes

121 - Software Engineering
PLO13. Know and apply methods of algorithm development, software design and data and knowledge structures.
PLO17. Be able to apply methods of component software development.
PLO18. Know and be able to apply information technology processing, storage and transmission of data.
PLO23. Be able to document and present the results of software development.

122 - Computer Science and Intelligent Systems
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.
PLO10. Use tools for developing client-server applications, design conceptual, logical, and physical models of databases, develop and optimize database queries, create distributed databases, repositories and showcases of databases, and knowledge bases, including those based on cloud services, using web programming languages.

Student workload

The total volume of the course is 120 hours (4 ECTS credits): lectures - 16 hours, laboratory classes - 32 hours, self-study - 72 hours.

Course prerequisites

121 - Software Engineering
Fundamentals of programming
Theory of algorithms
Object-oriented programming
Database design and development
Fundamentals of web development
Java basic programming course

Java advanced programming course

122 - Computer Science and Intelligent Systems
Algorithmization and programming
Algorithms and data structures
Databases
Object-oriented programming
Java basic programming course
Java advanced programming course
Fundamentals of web development

Features of the course, teaching and learning methods, and technologies

Teaching and learning methods
The main teaching method during lectures is the explanatory and illustrative method. In order to activate cognitive activity, student performances and the organization of discussions on specific issues of the lectures are provided. The course of laboratory works involves the development of Java code that allows you to solve typical tasks from the areas of Data Science.

Forms of assessment
Assimilation of the theory is tested in the form of an express survey during the lectures (CAS), a survey or an automated test at the beginning of the laboratory work (CAS).
Control of assimilation of material for independent study involves the preparation and defense of abstracts on an individual topics (2 abstracts) (CAS).
The level of practical skills is tested in laboratory works carried out according to individual options (CAS).
The final/semester control is carried out in the form of a credit, which involves the development of code in Java, which involves the application of learned technologies and methods on an individual task in a limited time (FAS).

Program of the course

Topics of the lectures

Topic 1
Data Science problems that can be solved in Java. Visualization of data in the form of documents, tables, diagrams.
Topic 2
Statistical and analytical processing of data arrays using Java.
Topic 3:
Organization of parallel processing of big data using Java tools.
Topic 4:
Organization of Machine Learning by means of Java.

Topics of the workshops

Workshops are not provided within the discipline.

Topics of the laboratory classes

Topic 1
Research of features of data visualization in the form of documents, tables, diagrams in Java.
Topic 2
Research of the peculiarities of solving data array processing problems by means of Java.
Topic 3
Research of features of parallel processing of big data by means of Java.
Topic 4
Research of the peculiarities of the implementation of Machine Learning by means of Java.

Self-study

Directions of Data Science research. Libraries and methods for exporting/importing data from Microsoft Office. Libraries, classes, methods for constructing graphs and diagrams. Libraries, classes, methods for statistical data processing using Java. Libraries, classes, methods for solving operations research problems using Java tools. Parallel calculations. Libraries, classes, methods for parallel processing of large data using Java tools. Typical tasks of Machine Learning. Libraries, classes, Machine Learning methods for Java tools.

Course materials and recommended reading

Key literature

1. Brzustowicz Michael. (2017) Data Science with Java: Practical Methods for Scientists and Engineers , O'Reilly Media, Inc., 236 p.
2. Haksun Li. (2020) Numerical Methods Using Java: For Data Science, Analysis, and Engineering, Apress LLC, 1180 p.
3. Raoul-Gabriel Urma, Mario Fusco, Alan Mycroft. (2018) Modern Java in Action. Lambdas, streams, functional and reactive programming, Manning, 532 p.
4. Малашонок Г. І., Сідько А. А. Паралельні обчислення на розподіленій пам'яті: OpenMPI, Java, Math Partner : підручник. – Київ : НаУКМА, 2020. – 266 с.
5. Ashish Singh Bhatia, Bostjan Kaluza. (2018)Machine Learning in Java: Helpful techniques to design, build, and deploy powerful machine learning applications in Java, 2nd Edition, 302 p.

Additional literature

Інтернет-ресурси

1. Створення графіків і діаграм у PDF на Java | Додайте лінію, прямокутник, коло тощо (aspose.com) https://blog.aspose.com/uk/pdf/create-graphs-and-charts-in-pdf-in-java/.
2. Створіть або створіть організаційну діаграму програмно в Java (aspose.com) // https://blog.aspose.com/uk/diagram/create-organizational-chart-java/
3. Apache POI – the Java API for Microsoft Documents // https://poi.apache.org/
4. Java 101: Understanding Java threads, Part 1: Introducing threads and runnables | JavaWorld // https://www.javaworld.com/article/2074217/java-101--understanding-java-threads--part-1--introducing-threads-and-runnables.amp.html.
5. Data Science and Machine Learning with Java | Udemy // https://www.udemy.com/course/data-science-and-machine-learning-with-java/.
6. https://www.baeldung.com/

Assessment and grading

Criteria for assessment of student performance, and the final score structure

The grade for the discipline consists of the points obtained during the course of study during the semester and the points awarded for the assessment. During the semester, a student can receive up to 80 points for:

- assimilation of theory (topics of independent work) (up to 30 points);
- performance of 4 laboratory works (up to 50 points).

A student can receive up to 20 points for completing assignments within a limited time frame.

Grading scale

Total points	National	ECTS
90–100	Excellent	A
82–89	Good	B
75–81	Good	C
64–74	Satisfactory	D
60–63	Satisfactory	E
35–59	Unsatisfactory (requires additional learning)	FX
1–34	Unsatisfactory (requires repetition of the course)	F

Norms of academic integrity and course policy

The student must adhere to the Code of Ethics of Academic Relations and Integrity of NTU "KhPI": to demonstrate discipline, good manners, kindness, honesty, and responsibility. Conflict situations should be openly discussed in academic groups with a lecturer, and if it is impossible to resolve the conflict, they should be brought to the attention of the Institute's management.
Regulatory and legal documents related to the implementation of the principles of academic integrity at NTU "KhPI" are available on the website: http://blogs.kpi.kharkov.ua/v2/nv/akademichna-dobrochesnist/

Approval

Approved by	08.06.2023	Head of the department Ihor HAMAIUN
	08.06.2023	Guarantors of the educational programs Andrii KOPP Uliya LITVINOVA