Course Syllabus

I. General Information

Course name	Fundamentals of algorithms and programming
Programme	Informatics/Mathematics
Level of studies (BA, BSc, MA, MSc, long-cycle MA)	BA
Form of studies (full-time, part-time)	Full-time
Discipline	Informatics, mathematics
Language of instruction	English

Course coordinator	Dorota Pylak, PhD

Type of class *(use only the types mentioned below)*	Number of teaching hours	Semester	ECTS Points
lecture	30	INF: II MAT: II,IV,VI	INF: 6 MAT: 5
tutorial			
classes			
laboratory classes	30	INF: II MAT: II,IV,VI	
workshops			
seminar			
introductory seminar			
foreign language classes			
practical placement			
field work			
diploma laboratory			
translation classes			
study visit			

Course pre-requisites	Introduction to computer science. Basic programming skills.

II. Course Objectives

Familiarizing students with the basics of Java programming
Presentation of the sorting and the searching algorithms
Introduction to the basics of object-oriented programming

III. Course learning outcomes with reference to programme learning outcomes

Symbol	Description of course learning outcome	Reference to programme learning outcome
KNOWLEDGE		
W_01	The student knows the syntax of the Java language: types, selection statements and loop statements	INF: K_W03, K_W06 MAT: K_W01, K_W04
W_02	The student knows how to define a simple class: its fields, constructors and methods.	INF: K_W01, K_W06 MAT: K_W01, K_W04
W_03	The student knows how to define one and two-dimensional arrays and how to write functions that operate on them.	INF: K_W01, K_W06 MAT: K_W01, K_W04
W_04	The student knows the operating principles and construction of recursive functions.	INF: K_W06 MAT: K_W01, K_W04
W_05	The student presents the basic sorting and searching algorithms.	INF: K_W01, K_W03, K_W06 MAT: K_W01, K_W04
W_06	The student knows the definitions of the enumeration type.	INF: K_W06 MAT: K_W01, K_W04
W_07	The student knows the basic applications of the Scanner, String classes.	INF: K_W01 MAT: K_W01, K_W04
SKILLS		
U_01	The student can write more advanced programs.	INF: K_U02, K_U04, K_U07, K_U08, K_U09, K_U11, K_U12, K_U17 MAT: K_U37
U_02	The student is able to create a functions that operate on the one and multidimensional arrays, select parameters and determine the result of the function.	INF: K_U02, K_U04, K_U11 MAT: K_U37
U_03	Student is able to create a recursive function for the given problem, can predict the result of the recursive function.	INF: K_U02, K_U04, K_U09 MAT: K_U37
U_04	Student is able to define a simple class, write a program that operates on the simple classes, and uses previously created functions.	INF: K_U02, K_U04, K_U11, K_U12 MAT: K_U37
U_05	The student can read data using the Scanner class.	INF: K_U02, K_U04, K_U11 MAT: K_U37
U_06	The student can use the enumeration type.	INF: K_U02, K_U04, K_U11 MAT: K_U37
U_07	The student is able to implement selected sorting and searching algorithms	INF: K_U02, K_U04, K_U07, K_U09 MAT: K_U37
SOCIAL COMPETENCIES		
K_01	The student is able to formulate a solution to the given problem, is open to the new solutions	INF: K_K01 MAT: K_K02
K_02	The student solves the given problems individually and while working in a group.	INF: K_K02 MAT: K_K02, K_K05

IV. Course Content

Introduction to Java. Types. Control statements.
Classes and introduction to object-oriented programming. Class definition, fields, methods, constructors.
One and two-dimensional arrays.
Recursive functions.
Basic sorting and searching algorithms.
The use of the recursion in sorting algorithms.
Input operations with Scanner.
Enumeration type.

V. Didactic methods used and forms of assessment of learning outcomes

Symbol	Didactic methods *(choose from the list)*	Forms of assessment *(choose from the list)*	Documentation type *(choose from the list)*
	KNOWLEDGE		
W_01	Conventional lecture / Guided practice	Exam/Written test	Examination card / written test
W_02	Conventional lecture / Guided practice	Exam/Written test	Examination card / written test
W_03	Conventional lecture / Guided practice	Exam/Written test	Examination card / written test
W_04	Conventional lecture / Guided practice	Exam/Written test	Examination card / written test
W_05	Conventional lecture / Guided practice	Exam/Written test	Examination card / written test
W_06	Conventional lecture / Guided practice	Exam/Written test	Examination card / written test
W_07	Conventional lecture / Guided practice	Exam/Written test	Examination card / written test
	SKILLS		
U_01	Practical classes design thinking	Exam/Written test	Examination card / written test
U_02	Practical classes design thinking	Exam/Written test	Examination card / written test
U_03	Practical classes design thinking	Exam/Written test	Examination card / written test
U_04	Practical classes design thinking	Exam/Written test	Examination card / written test
U_05	Practical classes design thinking	Exam/Written test	Examination card / written test
U_06	Practical classes design thinking	Exam/Written test	Examination card / written test
U_07	Practical classes design thinking	Exam/Written test	Examination card / written test
	SOCIAL COMPETENCIES		
K_01	Discussion, PBL (Problem-Based Learning) design thinking	Exam/Written test	Examination card / written test
K_02	Discussion, PBL (Problem-	Exam/Written test	Examination card /

	Based Learning) design thinking		written test

VI. Grading criteria, weighting factors.....

To pass a course, the student has to attend a classes and has to pass the tests and the final exam.

- passing classes - colloquia - 90% of the final grade, student's activity and work during classes - 10% of the final grade.

- written exam - for people who have passed the classes. Detailed conditions of exemption are given to students with each course edition.

Detailed assessment rules are given to the students with each edition of the course.

VII. Student workload

Form of activity	Number of hours
Number of contact hours (with the teacher)	**90**
Number of hours of individual student work	**60**

VIII. Literature

Basic literature
Herbert Schildt,Java: The Complete Reference, Eleventh Edition,McGraw-Hill Education, 2018 Herbert Schildt, Java: A Beginner's Guide, Eighth Edition, McGraw-Hill Education, 2018 http://docs.oracle.com/javase/8/docs/ http://docs.oracle.com/javase/11/docs/ C. S. Horstmann, G. Cornell, Core Java Volume I – Fundamentals (10th Edition), Pearson Education, 2018 C. S. Horstmann, Java, Core Java, Volume II--Advanced Features, 11th Edition, Pearson Education, 2019
Additional literature
R. Sedgewick, K. Wayne, Algorithms, 4th ed., Addison-Wesley, Upper Saddle River, NJ, 2011. N. Wirth, Algorithms + Data Structures = Programs, Prentice-Hall 1976