

Subject card

| | | | | | | | |
|--|---|--|------------------|-------------------------------------|---|------------|-----|
| Subject name and code | Programming Languages II, PG_00204161 | | | | | | |
| Field of study | Informatics | | | | | | |
| Date of commencement of studies | October 2026 | Academic year of realisation of subject | | | 2026/2027 | | |
| Education level | Bachelor's studies | Subject group | | | Obligatory subject group in the field of study Subject group related to practical vocational preparation | | |
| Mode of study | full-time studies | Mode of delivery | | | at the university | | |
| Year of study | 1 | Language of instruction | | | Polish | | |
| Semester of study | 2 | ECTS credits | | | 4.0 | | |
| Learning profile | practical | Assessment form | | | credit | | |
| Conducting unit | | | | | | | |
| Name and surname of lecturer (lecturers) | Subject supervisor | | dr Jakub Neumann | | | | |
| | Teachers | | | | | | |
| Lesson types | Lesson type | Lecture | Tutorial | Laboratory | Project | Seminar | SUM |
| | Number of study hours | 30.0 | 0.0 | 30.0 | 0.0 | 0.0 | 60 |
| | E-learning hours included: 0.0 | | | | | | |
| Learning activity and number of study hours | Learning activity | Participation in didactic classes included in study plan | | Participation in consultation hours | | Self-study | SUM |
| | Number of study hours | 60 | | 0.0 | | 40.0 | 100 |
| Subject objectives | The aim of the course is to learn and improve programming techniques, with particular emphasis on functional and object-oriented programming techniques and working with asynchronous code. | | | | | | |

| | | | |
|--|--|---|-----------------------------------|
| Learning outcomes | Course outcome | Subject outcome | Method of verification |
| | [INFPL3_U04] is able to use the acquired knowledge when creating, running and testing programs using dedicated tools and design patterns | can write a program in JavaScript, test its correctness in an integrated development environment, can optimize the code using tools such as linter and the lodash library | [SU4] test/exam - oral or written |
| | [INFPL3_K02] is ready to recognize the importance of knowledge in solving cognitive problems and practical and seeking opinions experts in case of difficulties with independent problem solving | can use a specialized language to take part in a discussion on programming (especially in JavaScript), present solutions to tasks during a code review | [SK4] test/exam - oral or written |
| | [INFPL3_W04] knows and understands advanced issues in programming, algorithms and computational complexity, programming languages and paradigms, as well as the complex relationships between these areas | has knowledge of object-oriented and functional programming in JavaScript, understands the principles of reasoning and working with asynchronous code | [SW4] test/exam - oral or written |
| | [INFPL3_U09] is able to - in accordance with the given specification - design and implement IT system | can write a program according to specifications given using a specialized language (concepts from the areas of functional and object-oriented programming in JavaScript and concepts related to asynchrony) | [SU4] test/exam - oral or written |
| [INFPL3_U08] is able to assess the usefulness of various paradigms and programming tools for solving various types of problems | in particular, understands the operations of reduce, map, filter, knows what function composition is and knows the concept of curry in relation to functions, is able to work with asynchronous code and use the Promise class | [SU4] test/exam - oral or written | |
| Subject contents | <p>Using JavaScript as an example:</p> <ul style="list-style-type: none"> functional programming techniques: higher order functions, functions returning functions, lambda expressions, currying definitions of reduce, map, filter functions, function composition creating objects defining classes, inheritance asynchrony and data types related to asynchrony lodash libraries, axios tools like linter, babel | | |
| Prerequisites and co-requisites | passed course Wstęp do programowania | | |
| Assessment methods and criteria | Subject passing criteria | Passing threshold | Percentage of the final grade |
| | tests | 51.0% | 100.0% |
| Recommended reading | Basic literature | JavaScript. Przewodnik. Poznaj język mistrzów programowania. aut. David Flanagan, ISBN 9788383227580 | |
| | Supplementary literature | Composing Software. An Exploration of Functional Programming and Object Composition in JavaScript, Eric Elliott | |
| | eResources addresses | | |
| Example issues/ example questions/ tasks being completed | | | |
| Work placement | Not applicable | | |

Document generated electronically. Does not require a seal or signature.