GDAŃSK UNIVERSITY OF TECHNOLOGY

Subject card

Subject name and code	Programming languages, PG_00045303		
Field of study	Data Engineering		
Date of commencement of studies	October 2024	Academic year of realisation of subject	2025/2026
Education level	first-cycle studies	Subject group	Obligatory subject group in the field of study Subject group related to scientific research in the field of study
Mode of study	Full-time studies	Mode of delivery	at the university
Year of study	2	Language of instruction	English
Semester of study	3	ECTS credits	2.0
Learning profile	general academic profile	Assessment form	assessment
Conducting unit	Department of Algorithms and Systems Modelling -> Faculty of Electronics, Telecommunications and Informatics		
Name and surname of lecturer (lecturers)	Subject supervisor	dr inż. Piotr Mironowicz	
	Teachers	dr inż. Piotr Mironowicz	

Lesson types and methods of instruction

Lesson type	Lecture	Tutorial	Laboratory	Project	Seminar	SUM
Number of study hours	15.0	0.0	15.0	0.0	0.0	30

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	30	4.0	16.0	50

Subject objectives

The aim of the course is an introduction to popular programming paradigms and getting the skill of their practical implementations.

Learning outcomes

Course outcome	Subject outcome	Method of verification

Subject contents	1. Procedural programming. 2. Linear syntax. FORTRAN 3. Activation records and subroutines 4. Recursive procedure call. 5. Block syntax. Control flow abstraction. 6. Binding of the names with objects. Range bonds. 7. The parameters of the procedure call. 8. Activation records for languages with recursion. 9. Static and dynamic calls. ALGOL. PASCAL. 10. Restrictions of block languages. 11. Abstraction of data and access protection. 12. Modularization. Modula-2. ADA83, ADA95 13. Exceptions. Exception handling models. 14. Concurrent procedures. Rendezvous. 15. Object-oriented programming. Objects, classes, hierarchies. 16. Dynamic types. Polymorphism. Smalltalk. C + +. 17. Recursive interpreted commands. 18. Symbolic transformation. Tail recursion. 19. Functional programming paradigm. 20. Haskell. Basic constructions and operations, module creation, performance. 21. Tacit programming. 22. LISP. General outline. 23. Prolog as an example of programming in logic. 24. Defining languages. Chomsky's classification, compilers. 25. Monads.

<table>
<tr><td>Prerequisites and co-requisites</td><td colspan="2"></td></tr>
<tr><td>Assessment methods and criteria</td><td colspan="2"><table>
<tr><th>Subject passing criteria</th><th>Passing threshold</th><th>Percentage of the final grade</th></tr>
<tr><td>tests</td><td>50.0%</td><td>40.0%</td></tr>
<tr><td>project</td><td>50.0%</td><td>60.0%</td></tr>
</table></td></tr>
<tr><td rowspan="3">Recommended reading</td><td>Basic literature</td><td>1. S. Mangano: XSLT receptury, wyd.2, Helion 2007

2. Cincom Smalltalk Downloads, http://www.cincomsmalltalk.com/

3. SAXON - The XSLT and XQuery Processor, http:// saxon.sourceforge.net/

4. W.F. Clocksin, W.F., Mellish, C.S.: Prolog Programowanie. Helion 2003

5. Ada Programming, http://en.wikibooks.org/wiki/Ada

6. SWI-Prolog downloads, www.swi-prolog.org/download.html

7. ADA Core, the GNAT Pro Company, http://www.adacore.com/home, https://libre.adacore.com/

8. D. S. Touretzky: Common Lisp: A Gentle Introduction to Symbolic Computation,

http://www.cs.cmu.edu/~dst/LispBook/

9. Z. Huzar, Z. Fryźlewicz, I. Dubielewicz, B. Hnatk: Ada 95, Helion 1998

10. Polski serwis języka Smalltalk, http://www.objectspace.net/</td></tr>
<tr><td>Supplementary literature</td><td>http://en.wikipedia.org/wiki/Programming_paradigm</td></tr>
<tr><td>eResources addresses</td><td>Adresy na platformie eNauczanie:</td></tr>
<tr><td>Example issues/ example questions/ tasks being completed</td><td colspan="2"></td></tr>
<tr><td>Work placement</td><td colspan="2">Not applicable</td></tr>
</table>

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