University of Gdańsk

Subject card

Subject name and code	Programming, PG_00159197		
Field of study	Quantum Information Technology		
Date of commencement of studies	October 2024	Academic year of realisation of subject	2024/2025
Education level	postgraduate studies	Subject group	Obligatory subject group in the field of study
Mode of study	full-time studies	Mode of delivery	at the university
Year of study	1	Language of instruction	English
Semester of study	1	ECTS credits	3.0
Learning profile	academic	Assessment form	
Conducting unit			
Name and surname of lecturer (lecturers)	Subject supervisor	dr inż. Piotr Mironowicz	
	Teachers	Fernando Almaguer Angeles dr inż. Piotr Mironowicz	

Lesson types

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	0.0	15.0	45

Subject objectives

The aim of this course is to provide a student a comprehensive overview of programming methodology that can be useful in further independent research in quantum information

Learning outcomes

Course outcome	Subject outcome	Method of verification
[QITL3_W05] knows the theoretical foundations of computational methods and IT techniques used to model and simulate physical systems considered in quantum information theory	Student knows computational complexity classes and formal programming structures	[SW4] test/exam - oral or written [SW5] implementation of a problem task
[QITL3_W02] has in-depth knowledge in the field of advanced mathematics and mathematical and computer methods, necessary to solve physical problems of medium complexity, and advanced knowledge in the area of quantum information and its technological aspects	Student can program in Python, Matlab, C++	[SW4] test/exam - oral or written [SW5] implementation of a problem task

<table>
<tr><td>Subject contents</td><td>Review and systematics of programming languages. Imperative and declarative programming. History and labor market. Programming environments.Program structure in C ++, Python, Matlab.Basic constructions. Variables, loops, conditional statements, functions, I / O operations, operators.Object-oriented programming. Classes. Basic data structures. Array, list, heap, map, graph.Code organization. Comments, headers, libraries, naming conventions. Programming Pragmatics. Programming styles. Version control systems.Doxygen.Recursion. Dynamic programming. Basic algorithms. Searching, sorting, graph searching.STL library in C ++. Design patterns. Processes and threads. Multi-threaded programming. Data Representations. XML. Sparse matrices. COO andCRS formats.Functional programming.Numerical Methods. Newton-Raphson method, Simpson method, Runge-Kutta method, matrix decompositions.Numpy and scipy packages in Python. Matlab QETLAB package.Linear and semi-definite programming. Solvers.Computational models. Turing machine. Church's thesis. Computational and memory complexity of algorithms. Complexity classes P, NP, NPC,PSPACE. Compilation process and parameters. Debugging and profiling. Unit tests.Code optimization techniques. Language interoperability. MEX files in Matlab. Extension modules in Python.CISC and RISC architectures. Flynn taxonomy. MMX, SSE, AVX instruction sets. Programming on graphic cards. CUDA, PyTorch.Virtual machines and emulators. Bytecode in Python. Assembler and low-level code optimization.BPP, BQP, QMA complexity classes. Quantum programming languages</td></tr>
<tr><td>Prerequisites and co-requisites</td><td>None.</td></tr>
<tr><td>Assessment methods and criteria</td><td>
<table>
<tr><th>Subject passing criteria</th><th>Passing threshold</th><th>Percentage of the final grade</th></tr>
<tr><td>laboratory part: test</td><td>51.0%</td><td>50.0%</td></tr>
<tr><td>lecture part: test</td><td>51.0%</td><td>50.0%</td></tr>
</table>
</td></tr>
<tr><td>Recommended reading</td><td>
<table>
<tr><td>Basic literature</td><td>None.</td></tr>
<tr><td>Supplementary literature</td><td>None.</td></tr>
<tr><td>eResources addresses</td><td>Podstawowe
https://octave.org/octave.pdf - GNU Octave Free Your Numbers – reference manual
https://docs.python.org/3/index.html - Python3 Documentation
http://www.cplusplus.com/reference/ - C++ Reference
https://www.mathworks.com/help/matlab - Matlab Reference Manual
Adresy na platformie eNauczanie:</td></tr>
</table>
</td></tr>
<tr><td>Example issues/ example questions/ tasks being completed</td><td></td></tr>
<tr><td>Work placement</td><td>Not applicable</td></tr>
</table>

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