COURSE DESCRIPTION CARD - SYLLABUS

Course name

Selected applications of quantum computers

Course

Field of study	Year/Semester
Technical Physics	4/7
Area of study (specialization)	Profile of study
	general academic
Level of study	Course offered in
First-cycle studies	polish
Form of study	Requirements
full-time	elective

Number of hours

Lecture	Laboratory classes	Other (e.g. online)
30		
Tutorials	Projects/seminars	

Number of credit points

4

Lecturers

Responsible for the course/lecturer:

dr Gustaw Szawioła

Wydział Inżynierii Materiałowej i Fizyki Technicznej

Piotrowo 3, 60-965 Poznań

gustaw.szawiola@put.poznan.pl

Responsible for the course/lecturer:

Prerequisites

Knowledge and skills in the field of quantum physics, the basics of quantum engineering as well as mathematical and programming competences specified in the learning outcomes for first-cycle and second-cycle studies (1st and 2nd semester) in the field of technical physics.

Course objective

The module presents the structure, practical implementations of selected quantum algorithms, as well as the perspective of their practical application.

Course-related learning outcomes

Knowledge

1. The student, using adequate concepts and methods of quantum informatics, explains the logical

structure and limitations of selected quantum algorithms (specified in the program content). [K2_W01, K2_02, K2_W07, K2_W10].

2. The student explains the details of the practical application and implementation of the selected quantum algorithm in Python, with the use of open quantum libraries. [K2_W07, K2_W10].

Skills

1. The student plans to implement the indicated quantum algorithm in Python with the use of open quantum libraries. [K2_U01, K2_U04, K2_U05, K2_U07]

2. The student implements the indicated quantum algorithm on a quantum simulator (or a quantum computer available in the cloud), which is coded in Python with the use of open quantum libraries. [K2_U01, K2_U05, K2_U12, K2_U21.

3. The student carries out selected physical tests of the quantum computer. [K2_U01, K2_U14, K2_U16]

Social competences

1. The student is aware of the relationship between the dynamics of information technology development and the achievements of quantum engineering. [K2_K04, K2_K08]

2. The student conscientiously and ethically fulfills the duties, demonstrating a constructive attitude in the undertaken discussions. [K2_K03, K2_K07]

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Knowledge (W01, W02) - written exam in the form of a test (W) with a maximum number of points Wmax = 12p

Skills (U01, U02, U03) - two home programming control tasks (U) with a maximum number of points for two tasks Umax = 10 points (5 points for a task), the completion of tasks is verified in the form of an oral defense during the oral exam.

Social competences (K01, K02) -Kmax = 3p, objectivity and argumentation during the oral exam are assessed.

Programme content

I Fundamentals of quantum computer programming in Python.

I.1. Open quantum programming platforms, programming languages and programming environments - an overview. Configuration of the development environment.

I.2. Review of Python syntax and key quantum libraries.

I.3. Implementation of basic quantum gates and basic quantum functional circuits in Python. An example of simulation of a selected quantum communication protocol.

I.4. The problem of synthesis and decomposition of quantum circuits. The universality of quantum gates. Transpilation of quantum circuits.

II. "Canon" of quantum algorithms

II.1 Implementation of the following algorithms: Deutsch-Jozs, Bernstein-Vazirani, Simon.

II.2. Implementation of the quantum Fourier transform algorithm.

II.3. Implementation of the quantum phase estimation algorithm.

II.4. Implementation of the Shor algorithm.

III.Selected Applications.

III.1. Quantum algorithm for solving linear equations.

III.2. Quantum algorithm for determining eigenvalues using the variation method - application to molecule simulation.

III.3. Simulation of quantum dynamics. The Suzuki-Trotter approximation

III.4 Quantum random walk.

IV. Qubit quantum hardware.

IV.1 Hardware determinants of quantum computing.

IV.2 Comparative tests of quantum computers (quantum benchmarks).

Teaching methods

Lecture: multimedia presentation illustrated with examples of implementation of quantum algorithms..

Bibliography

Basic

1. A. Asfaw, i in. , ,,Learn Quantum Computation Using Qiski, (2020) http://community.qiskit.org/textbook

2. J.D. Hidary ,,Quantum Computing: An Applied Approach'' Springer , 2019

3. M. Lutz, "Python. Wprowadzenie. Wydanie V", Helion 2020

Uzupełniająca

1. M. Senekane, ,,Hands-On Quantum Information Processing with Python'', Packt Publishing, 2021

2. Python Tutorial, https://www.w3schools.com/python/default.asp

Additional

1. M. Senekane, ,,Hands-On Quantum Information Processing with Python'', Packt Publishing, 2021

2. Python Tutorial, https://www.w3schools.com/python/default.asp

Breakdown of average student's workload

	Hours	ECTS
Total workload	68	4
Classes requiring direct contact with the teacher	38	2,0
Student's own work (literature studies, preparation for laboratory classes/tutorials, preparation for tests/exam, project preparation) [1]	30	2,0

[1] delete or add other activities as appropriate