Quantum algorithms and programming

CS101-3

9 units (3-0-6), P/F. Spring term. Prerequisites: CS 1, CS 21, CS 38, Ma 1b., or equivalent.
Instructor: Alexandru Gheorghiu

Outline. This course will teach the fundamentals of quantum algorithms through a combination of theory and programming. The goal will be to understand how quantum algorithms work, what advantages they offer over non-quantum (classical) algorithms, as well as the limitations of efficient quantum computation. On the theory side, this will be done using tools from algorithms analysis and computational complexity theory. On the programming side, we will be implementing a number of algorithms in the quantum programming language Q#. The course will cover standard quantum algorithms, such as those of Deutsch-Jozsa, Simon, Shor and quantum simulation, as well as some of the more modern ones, such as algorithms for near-term devices and quantum machine learning. The main programming languages used in the course will be Python and Q#. Students should have basic familiarity with Python. No prior knowledge of Q# is required, though students will be expected to learn it throughout the course. The course will introduce the basics of quantum computation and so no prior knowledge of the subject is required.

Prerequisites. Students taking this course should be familiar with the *basics* of linear algebra, programming, algorithms and data structures. While the course will also serve as an introduction to quantum computation, students that have completed a quantum computing course (such as Ph 219), and wish to deepen their understanding of quantum algorithms are encouraged to take this course.

Grading. Students will be graded on four homework assignments and one group project as follows:

15% *Homework 1.* Programming assignment in Python.

15% *Homework 2.* Programming assignment in Q#.

15% *Homework 3.* Written assignment.

15% *Homework 4.* Programming assignment in Q#.

40% *Group project.* A team-based assignment in which students will be asked to do a quantum algorithms project in their programming language of choice and write a short report (under 10 pages) on the project and the obtained results. The team will also give a short presentation of their project in the last lecture.

As the course is P/F, students are required to complete at least 50% in order to pass.