

ECSE 4964: Quantum Computer Programming

Department of Electrical, Computer, and Systems Engineering
Rensselaer Polytechnic Institute

Course Information

Course Title: Quantum Computer Programming
Transcript Title: Quantum Computer Programming
Course Number: ECSE 4964
Semester and Year: Fall 2020
Credit Hours: 3
Meeting Times: M R 4:45-6:05 PM
Classroom: online, via WebEx

Instructor

Prof. W Randolph Franklin
Office: JEC 6026
Office Phone: x6077
Email: frankwr@rpi.edu
Office Hours: after class and by appointment, via WebEx, email or phone.
TA(s): none

Course Description

Intro to quantum mechanics. Various physical realizations of quantum computing. The IBM Q. Quantum states and qubits. Quantum gates including Hadamard, Pauli-XYZ, Toffoli, Fredkin. Qiskit. Quantum algorithms, including Grover, Shor, and recent quantum algorithms. Investigating quantum hardware using qiskit.

Pre-requisites

ECSE 2610 and CSCI 2200 and PHYS 1200.

Textbooks

- 1 Noson S. Yanofsky and Mirco A. Mannucci, *Quantum Computing for Computer Scientists*, 2008. Etext \$60.
- 2 Abraham Asfaw et al, *Learn Quantum Computation using Qiskit*, <http://qiskit.org/textbook>, 2020. Free.
- 3 Rishwi Thimmaraju and Harika Vajha. *Beyond Classical: A crash course on Quantum Computing using Qiskit and IBM*, 2020. Kindle: \$7 or paperback: \$9.

Student Learning Outcomes

Students who successfully complete this course will be able to

- 1 Demonstrate proficiency with the mathematics behind quantum algorithms.
- 2 Identify the concepts with today's non-fault-tolerant quantum devices.
- 3 Apply these procedures to write code in Qiskit to implement quantum algorithms on IBM's cloud quantum systems.

Course Assessment Measures

- 1 Course assessment will be done through homeworks, class presentations, and a final project.

- 2 There will be approximately five homeworks. They may be done in teams of two students.
- 3 There will be two 10-minute slide presentations or videos, to be produced by teams of two students and shown in class. For the presentations, the students will select some relevant topic in quantum computing from a list suggested by the instructor, and present it to the class. Each group of presentations may take two class days.
- 4 The final project may be done in teams of two to four students. Deliverables will include a report and a video presentation, modeling a conference paper and presentation. The report will be in the form of a scientific conference paper. It will be about eight formatted pages long. Showing the presentations will fill the last two class days.
- 5 Gradescope will be used to manage the grading process.
- 6 The instructor will use a static content management system on his private virtual web server at <https://wrf.ecse.rpi.edu/nikola/pages/Teaching/quantum-f2020/> to maintain an online blog containing syllabus, homeworks, and lecture summaries.
- 7 Some programming assignments will use IBM quantum computing simulator that is available on github. That may be downloaded and run on any machine, such as a student's personal machine. For more serious computation, parallel.ecse.rpi.edu is available, but probably not necessary. It is a dual 14-core Intel Xeon with 256GB of main memory.
- 8 Other assignments will use IBM's teaching quantum computer, which is available on the web. This will allow the students to see that the simulator is not perfect.
- 9 When the final numerical grade is converted to a letter, if a student has participated enthusiastically and positively in class and on piazza, that may bump the letter grade up into the next category.

Grading policy

- | | |
|--------------------------------------|---|
| 1 Homework assignments | 50% |
| 2 Two in-class presentations | 10% each |
| 3 Final project (paper/presentation) | 30% |
| 4 Piazza and class participation | might push the letter grade up into the next category |

Course Policies

- 1 Lectures will be recorded (subject to WebEx failures) and be available to students. Questions will be encouraged, and included in the recording.
- 2 Students are encouraged to attend in real time rather than only watching the recording later.
- 3 Late homeworks will be allowed in a case-by-case basis.
- 4 Late projects will trigger RPI's incomplete-grade rules, with all that implies.
- 5 Students who discover a nontrivial error by the instructor will receive a bonus.
- 6 Students who provide extra material that the instructor uses will receive a bonus.
- 7 As this is a reading, writing, talking, and programming course, there are no lab safety issues.
- 8 Homeworks, and projects are encouraged to build on existing material that the student has legal access to, and which is acknowledged.
- 9 The in-class presentations will be summaries of intellectual material, such as conference presentations, created by others, who will be acknowledged.
- 10 There will be a piazza site for students to post questions to other students and the prof.
- 11 A small number of points will be awarded for participation.
- 12 Homeworks will be submitted on gradescope. Students must agree to allow gradescope to store the data (FERPA).
- 13 This course is attempting to use several different computer systems together. When they are good, they are very good indeed, but when they are bad, they are horrid. So, system failures may force changes to the above rules.

Sample Course Topics and Tentative Weekly Schedule

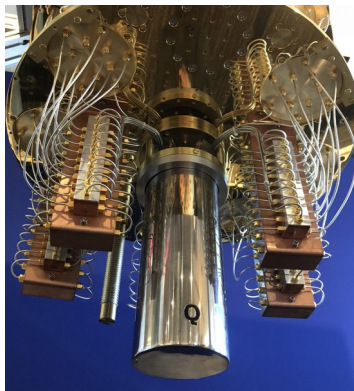
- 1 Intro to quantum mechanics, Introduction to Python and Jupyter notebooks.
- 2 Various physical realizations of quantum computing, Quantum states and qubits
- 3 Quantum gates, Qiskit
- 4 Student in-class presentations, round 1
- 5 quantum algorithms, Grover's algorithm
- 6 Superdense Coding and Quantum Teleportation
- 7 Shor's algorithm, Quantum Algorithms for Applications
- 8 Implementations of Recent Quantum Algorithms
- 9 The Variational Quantum Linear Solver
- 10 Student in-class presentations, round 2
- 11 Investigating quantum hardware using qiskit
- 12 Simulating Molecules using the Variational Quantum Eigensolver (VQE)
- 13 Introduction to Quantum Error Correction via the Repetition Code
- 14 Final presentation

Academic Integrity

Student-teacher relationships are based on trust. For example, students must trust that teachers have made appropriate decisions about the structure and content of the courses they teach, and teachers must trust that the assignments that students turn in are their own. Acts which violate this trust undermine the educational process. Please refer to the *Rensselaer Handbook of Student Rights and Responsibilities and The Rensselaer Graduate Student Supplement* for definitions of various forms of academic dishonesty and the applicable penalties.

Specifics for this course are as follows.

- 1 You may collaborate on homeworks, but each team must write up the solution separately (one writeup per team) using their own words. We willingly give hints to anyone who asks.
- 2 The penalty for two teams handing in identical work is a zero for both.
- 3 You may collaborate in teams of up to 4 people for the term project.
- 4 You may get help from anyone for the term project. You may build on a previous project, either your own or someone else's. However you must describe and acknowledge any other work you use, and have the other person's permission, which may be implicit. E.g., my web site gives a blanket permission to use it for nonprofit research or teaching. You must add something creative to the previous work. You must write up the project on your own.
- 5 However, writing assistance from the Writing Center and similar sources is allowed, if you acknowledge it, and it does not provide an intellectual contribution.
- 6 The penalty for plagiarism is a zero grade.
- 7 Cheating will be reported to the Dean of Students Office.



IBM Q