ECSE 4964/6964: 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 4964/6964
Semester and Year: Fall 2022
Credit Hours: 3
Meeting Times: M R 4-5:20 PM
Web page: https://wrfranklin.org/Teaching/quantum-f2022/

Instructor

W Randolph Franklin, Professor, ECSE
Office: JEC 6026
Office Phone: 518-276-6077
Email: mail@wrfranklin.org or 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, such as transmon qubit (IBM Q), trapped ion (IonQ), and quantum annealing (D-Wave). Quantum states and qubits. Quantum gates including Hadamard, Pauli-XYZ, Toffoli, Fredkin. Qiskit. Quantum algorithms such as Grover, and Shor. Programming quantum computers using IBM qiskit and Microsoft Quantum.

Pre-requisite

ECSE 2610 (Computer Components and Operations) and CSCI 2200 (Foundations of Computer Science*),* and PHYS 1200 (Physics II), or permission of the instructor. Which is usually granted.

Suggested Textbooks

1 Jack Hidary: Quantum computing: an applied approach. The PDF is $30 from https://www.springer.com/us/book/9783030239213 That is cheaper than Amazon. The affiliated site is https://github.com/jackhidary/quantumcomputingbook
2 Michael A. Nielsen and Isaac L. Chuang, Quantum Computation and Quantum Information: 10th Anniversary Edition, "Mike and Ike". https://www.amazon.com/Quantum-Computation-Information-10th-Anniversary-ebook-dp-B07FPFL6HG/dp/B07FPFL6HG?me= , kindle $46
3 Abraham Asfaw et al, *Learn Quantum Computation using Qiskit,* http://qiskit.org/textbook, 2020. Free.
4 A curated collection of web material, free.

Student Learning Outcomes

Students who successfully complete this course will be able to

1. Demonstrate proficiency with the mathematics behind quantum algorithms, such as unitary operators and quantum gates.
2. Understand the major quantum computing algorithms, such as Grover and Shor.
3. Compare and contrast the three main hardware platforms: transmon qubit (IBM Q), trapped ion (IonQ), and quantum annealing (D-Wave).

In addition to 3 outcomes above, students who successfully complete the 6000-level version of this course will be able to

Apply all the quantum computing algorithms to create and run programs on the three main platforms

Course Assessment Measures

1. Course assessment will be done through homeworks, class presentations, and a final project.
2. There will be many small homeworks. They may be done in teams of two students.
3. There will be several 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.
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-f2022 to maintain an online blog containing syllabus, homeworks, and lecture summaries.
7. Some programming assignments will use, e.g., the 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 all three major quantum architectures, which are available on the web, e.g., from IBM, Microsoft, and Amazon.
9. When the final numerical grade is converted to a letter, if a student has participated enthusiastically and positively in class, that may bump the letter grade up into the next category.
10. All the preceding may be modified for good reason. E.g., if there are more students, then we may have fewer student presentations (per student).

In addition for the 6000-level version, students will be required to

11. pick a major current topic in quantum computing, and write a review of its state, formatted as for a published journal paper.

Grading policy

4000 level:

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

6000 level:

1.	Homework assignments	30%
2.	One or two in-class presentations	10% each
3.	Final project (paper/presentation)	30%
4.	Major survey paper	20%
5.	Class participation	might push the letter grade up into the next category

The number of in-class presentations will depend on the number of students. If there are too many students, then we will reduce the number of presentations from two to one, and renormalize the weights.

Course Policies

1. Students may not take both 4964 and 6964 for credit.
2. This is intended to be an in-person course that will not be recorded. Future events might modify this.

3. Students are expected to attend in person.
4. Late homeworks will be allowed in a case-by-case basis.
5. Late projects will trigger RPI's incomplete-grade rules, with all that implies.
6. Students who discover a nontrivial error by the instructor will receive a bonus.
7. Students who provide extra material that the instructor uses will receive a bonus.
8. As this is a reading, writing, talking, and programming course, there are no lab safety issues.
9. Homeworks, and projects are encouraged to build on existing material that the student has legal access to, and which is acknowledged.
10. If you use the same work for two courses, you are required to say so, and to convince both profs that you did extra.
11. The in-class presentations will be summaries of intellectual material, such as conference presentations, created by others, who will be acknowledged.
12. A small number of points will be awarded for participation.
13. Homeworks will be submitted on gradescope. Students must agree to allow gradescope to store the data (FERPA).
14. 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.
15. Students form a professional relationship with the instructor and are welcome to professional help like advice and recommendations even after graduation.
16. The 2022 version of this course will be mostly like last year, but tweaked to improve it.

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 qbits
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. Trapped ion quantum computers and IonQ
13. Quantum annealing and D-Wave
14. Final presentation

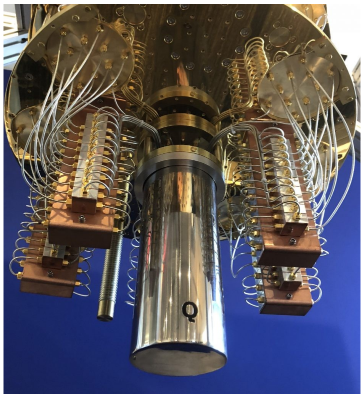
Academic Integrity

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.

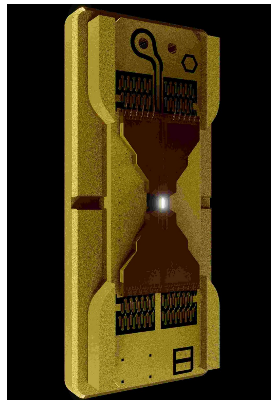
Specifics for this course are as follows.

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.

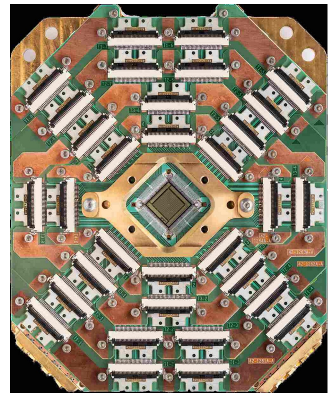
5. However, writing assistance from the Writing Center and similar sources is allowed, if you acknowledge it, and it does not provide an intellectual contribution.
6. The penalty for plagiarism is a zero grade.
7. Cheating will be reported to the Dean of Students Office.



IBM Q



IonQ



D-Wave