Quantum Software Development

EE-193 / CS-150 | Spring 2024 | Tufts University

Key Information

Instructor: Richard Preston

Instructor Email: rhpreston@mitre.org

Class Time: Wednesdays, 4:30PM - 7:00PM, Joyce Cummings Center Room 265

Office Hours: Wednesdays, 3:00PM - 4:30PM, Joyce Cummings Center Room 625;
Wednesdays, 7:00PM - 8:30PM, Joyce Cummings Center Room 402

Exam 1: Wednesday February 21, 4:30PM, Joyce Cummings Center Room 265

Exam 2: Wednesday May 8, 12:00PM, location TBD

Canvas: canvas.tufts.edu/courses/54025

Discord Server: discord.gg/PCjUJFPksX

Graduate Student Handbook: tufts.app.box.com/v/soe-grad-handbook

Attendance is mandatory.
Assignments are provided and submitted through Canvas.
Discord is used for all communication outside of class.

Course Description

Quantum computing is an emerging computing technology that uses the principles of quantum mechanics instead of digital logic to process information. Many algorithms designed to work on a quantum computer have been shown to provide dramatic computational advantage over classical algorithms, with relevant applications in cryptography, machine learning, simulation, optimization, and more. Recent advances in quantum computing hardware have demonstrated the feasibility of employing quantum computation to solve real-world problems. Industry road maps point to commercial availability of large-scale quantum devices in the next decade.

The objectives of this course are:

1. Understand quantum computation. Students should be able to correctly and clearly explain how quantum computers work at the level of information processing.
2. Attain competency in quantum software engineering. Students should be able to implement and study quantum algorithms using software.

Grades

The final grade reflects a student's completion of the stated objectives and mastery of the subject material. It is the weighted average of the following scores:

Participation	15%
Assignments	25%
Exams	30%
Final Project	30%

The letter grade is determined by the final score rounded to the nearest hundredth as follows:

A+	0.98 – 1.00
A	0.94 – 0.97
A-	0.91 – 0.93
B+	0.88 – 0.90
B	0.84 – 0.87
B-	0.81 – 0.83
C	0.71 – 0.80
D	0.61 – 0.70
F	< 0.60

The grading policy will not be changed without the consent of the students. Any deviations from the syllabus must be applied equally to all students.

Participation

Students are expected to show up for class and participate in the discussion. **A short quiz will be given at the beginning of most classes covering the material from the previous lecture.** Though the lecture slides will be available for review, students are not expected to spend time preparing for quizzes. Students earn one point for each quiz completed, and one additional point if they participate on the day the quiz is given. The total number of points is divided by 20 to calculate the final participation score, with a maximum score of 1.0.

Assignments

There are 9 coding assignments, with due dates specified in the calendar on the next page. **Assignments are due by class time (4:30PM), with an automatic extension to 11:59PM.** Collaboration with other students is encouraged, but acknowledgement of all collaborators must be provided. Each assignment is scored as the proportion of exercises completed correctly, with half credit given for incorrect attempts. Some assignments may contain extra credit, allowing for a score greater than 1.0. The final assignment score is the average of the individual scores.

Assignments submitted up to one week after the due date will be accepted for half credit. **Assignments more than one week late will not be accepted.** Exceptions may be made for extenuating circumstances, such as severe illness.

Exams

A mid-term and end-of-term written exam will be given on material covered up to that point in lecture. Exams may include multiple choice, short answer, and coding questions. A single notes sheet is allowed for reference, and ample time will be provided. Each exam is scored as the proportion of correct responses, with half credit given for incorrect attempts. The final exam score is the average of the two exam scores.

Final Project

The final project assignment is to write a software implementation of a quantum algorithm. Students will present their work, including a computational resource estimation, in a short video. **The final project is due at the end of Reading Period on May 3rd.** Failures to meet the deadline will be handled on a case-by-case basis; it is expected that students earnestly work to complete the project with expedition.

Calendar

Lecture topics may diverge from the calendar to maintain a pace that works well for the class. Assignment deadlines and exam dates are firm; they may be changed only for class cancellation or similar unexpected circumstances.

Date	Assignment / Exam	Lecture Topic(s)
1/17		Course Intro, Syllabus, Math Refresher
1/24		Q. Information, Q. Logic Gates, Intro to Q#
1/31	**Lab 1 Due**	Visualizing Single-Qubit States, Working with Multiple Qubits
2/7	**Lab 2 Due**	Quantum Control Logic
2/14	**Lab 3 Due**	Quantum Communication, Quantum Error Correction
2/21	**Lab 4 Due**	Quantum Interference, Midterm Review
2/28	**Exam 1**	
3/6		Basic Quantum Algorithms
3/13	**Lab 5 Due**	Hybrid Algorithms, The Multiverse
3/20		NO CLASS
3/27	**Lab 6 Due**	Grover’s Search Algorithm, Final Project Discussion
4/3	**Lab 7 Due**	Quantum Fourier Transform
4/10	**Lab 8 Due**	Shor’s Factorization Algorithm
4/17		MAKE-UP DAY
4/24	**Lab 9 Due**	Cloud Execution, Final Review
5/3	**Final Project Due**	
5/8	**Exam 2**	

Additional Resources

- Scott Aaronson’s Introduction to Quantum Information Science Lecture Notes: www.scottaaronson.com/qclec.pdf
- David Deutsch, The Beginning of Infinity (the audiobook is quite good): a.co/d/iT4qFfV
- Microsoft’s Quantum Katas: quantum.microsoft.com/en-us/experience/quantum-katas