Quantum Computing

Course Outline

Prachya Boonkwan

Version: January 17, 2026

Sirindhorn International Institute of Technology
Thammasat University, Thailand

Who? Me?

- Nickname: Arm (P'/N' Arm, etc.)
- Born: Aug 1981
- Work
  - Researcher at NECTEC 2005-2024
  - Lecturer at SIIT, Thammasat University 2025-now
- Education
  - B.Eng & M.Eng in Computer Engineering, Kasetsart University, Thailand
  - Obtained Ministry of Science and Technology Scholarship of Thailand in early 2008
  - Did a PhD in Informatics (AI & Computational Linguistics) at University of Edinburgh, UK from 2008 to 2013

Course Description

- This course provides a comprehensive introduction to quantum computing, bridging the gap between theoretical physics and practical engineering applications.
- Starting with the mathematical foundations of complex linear algebra, students will progress through the core principles of qubits, quantum logic gates, and circuit design.
- The curriculum covers essential quantum algorithms (including Shor's, Grover's, and HHL), error correction, and system dynamics. A significant portion of the course is dedicated to the emerging fields of Quantum Machine Learning (QML) and variational algorithms (VQE, QAOA).
- Through the sessions, students will gain hands-on experience simulating quantum circuits and algorithms using the PennyLane framework, preparing them to tackle problems in cryptography, optimization, and system simulation on Noisy Intermediate-Scale Quantum (NISQ) devices.

https://github.com/kaamanita/quantum-computing

Course Syllabus/1

Format: 15 sessions (2+1 hours)

1. Introduction & Complex Numbers (CH1)
   - Complex numbers, polar forms, wave representation, inner/outer products, phase synchronicity.
2. Complex Matrices (CH1)
   - Hermitian and Unitary matrices, eigen-decomposition, matrix exponentials, Householder reflection.
3. Qubits, Operators, & Bloch Sphere (CH2)
   - Probability distributions, Dirac notation, Bloch sphere visualization, Universal rotation, global phase vs. relative phase.

Tools: Python, PennyLane, generative AI

4. Quantum Logic & Entanglement (CH2)
   - Tensor products, multi-qubit systems, entanglement (Bell/GHZ/W-states), measurement bases.
5. Circuit Design & PennyLane Basics (CH3)
   - The Clifford Set (Levels 1-3), constructing circuits in PennyLane, QNode decorators, state preparation algorithms.
6. Measurement & Dynamic Circuits (CH4)
   - Analytical vs. Projective measurement, expectation values, variance. Introduction to dynamic circuits (loops, conditionals).

Course Syllabus/2

Format: 15 sessions (2+1 hours)

Tools: Python, PennyLane, generative AI

7. Non-Clifford Gates & Noise (CH4)
   - Toffoli gates, generalized rotation, density matrices, mixed states, and simulating noise/decoherence.
8. Foundational Algorithms (CH5)
   - Oracle separation, Phase Kickback technique. Bernstein-Vazirani and Deutsch-Jozsa algorithms.
9. Communication & Error Correction (CH5)
   - No-Cloning theorem, Teleportation, Superdense coding. Quantum Error Correction (Repetition codes, Shor code).
10. Grover's Search Algorithm & QFT (CH6)
    - Unstructured search, amplitude amplification, geometric interpretation, optimal iteration counts, Quantum Fourier Transform (QFT), Inverse QFT.
11. QPE & Cryptography (CH6)
    - Quantum Phase Estimation (QPE), logic behind Shor's Algorithm and RSA breaking.
12. System Dynamics & Automata (CH7)
    - Deterministic vs. Probabilistic vs. Quantum Finite-State Automata (QFA), Szegedy's algorithm, Quantum Walks.

Course Syllabus/3

Format: 15 sessions (2+1 hours)

13. Advanced Simulation (CH7)
    - Hamiltonian simulation, Trotter-Suzuki decomposition, HHL algorithm for linear systems, Quantum Signal Processing (QSP).
14. Quantum Machine Learning (CH8)
    - Variational Quantum Algorithms (VQA), Ansatz design (HEA, TEN, ALT), Quantum Support Vector Machines (QSVM), Quantum Neural Networks (QNN), gradients (Parameter Shift Rule).

Tools: Python, PennyLane, generative AI

15. Optimization & VQE (CH9)
    - Variational Quantum Eigensolver (VQE), QAOA for graph problems (Max-Cut, Max-Clique, Traveling Salesman Problem).

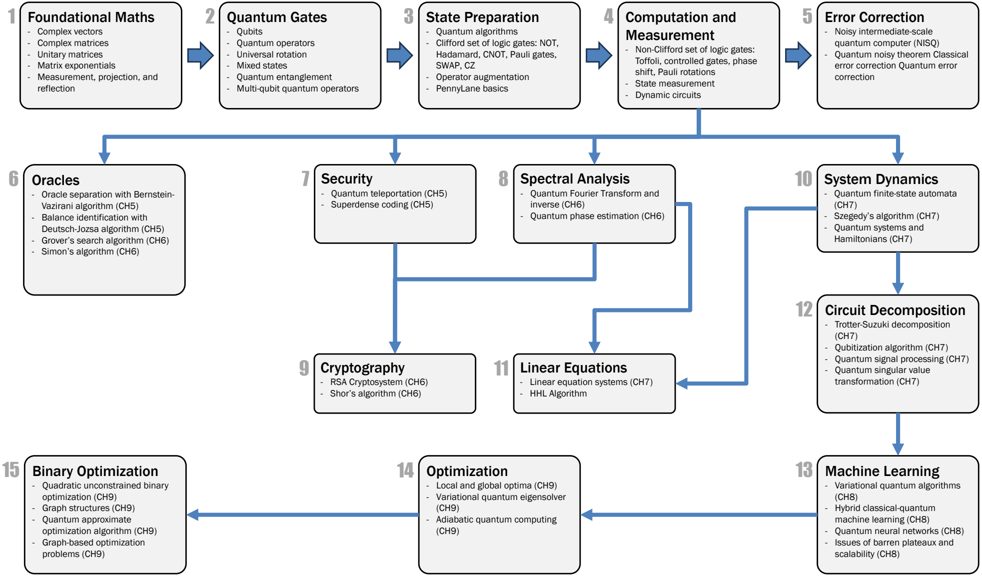

Figure 1: Course map and epistemological flow

Scoring and Exams

- Midterm exam (2 hours) and final exam (3 hours) are open-book and dictionary, but generative AI is still NOT allowed
- Two assignments are based on the exercises incorporated in the slides
- Two popup quizzes are done on Google Forms and the questions are quite easy

Midterm	30
Final	30
Assignments $\times$ 2	20
Popup quizzes $\times$ 2	20
Total	100

Table 1: Scoring

- You are allowed to use generative AI while doing assignments, but be aware that it frequently hallucinates when dealing with complex math equations! Even the pro version still does! Be vigilant …
- Midterm and final exams will be based on the target learning outcomes listed in the next slides, so be prepared
- Majority of my exam questions are generally conceptual, intended to assess your basic understandings
- But there may also be 1-2 analytical math questions, intended to assess your basic math skills

Supplementary Materials

- IBM SkillsBuild: Open Plan for Educators and Students in Quantum Computing
    - `https://skillsbuild.org/students/course-catalog/quantum-computing`
    - Once you register to the platform, you have to enroll to their online course: Quantum Enigmas Parts 1-2, to familiarize with the notion of quantum computing and IBM's platform
    - In most cases, you can finish the whole course in 1-2 days. Believe me, the materials are relatively easy and equipped with a generous amount of interactive visualizations!
    - Once you complete this course, you will obtain an access credential with 10 minutes/month of computing time, which is quite plentiful for many tasks

Program Learning Outcomes

PLO-1 (Theoretical Foundation): Apply principles of linear algebra (complex vectors, Hermitian/Unitary matrices, eigen-decomposition) to model quantum mechanical phenomena such as superposition and entanglement.

PLO-2 (Algorithmic Implementation): Construct and simulate quantum algorithms (including Grover's, Shor's, and HHL) using the PennyLane Python library to solve computational problems.

PLO-3 (Circuit Design): Design and optimize quantum circuits using Clifford and non-Clifford gate sets, dynamic circuits, and state preparation techniques to achieve specific quantum states.

PLO-4 (System Analysis): Analyze the behavior of quantum systems, including noise modeling, error correction, and system dynamics, to evaluate the robustness and fidelity of quantum computations.

PLO-5 (Advanced Application): Develop hybrid classical-quantum solutions for machine learning (QNN, VQE) and combinatorial optimization (QAOA) problems.

Target Learning Outcomes/1

TLO-1: Convert between standard and polar forms of complex numbers to represent wave amplitudes and calculate inner/outer products to determine phase synchronicity.

TLO-2: Identify Hermitian and Unitary matrices and perform eigen-decomposition to interpret quantum gates as rotations and projections.

TLO-3: Decompose quantum operators into universal rotations and represent multi-qubit systems using tensor products and density matrices.

TLO-4: Construct quantum circuits using the Clifford gate set and implement state preparation algorithms for Bell and GHZ states in PennyLane.

TLO-5: Implement mid-circuit measurements and dynamic logic (loops, conditionals) to control quantum algorithm flow.

TLO-6: Differentiate between analytical, projective, and expectation-based measurements to extract information from quantum systems.

Target Learning Outcomes/2

TLO-7: Implement oracle-based algorithms (Bernstein-Vazirani, Deutsch-Jozsa) to demonstrate quantum speedup and phase kickback.

TLO-8: Apply Quantum Error Correction codes (Shor code) and model noisy channels (bit-flip, depolarization) to mitigate decoherence.

TLO-9: Construct the Quantum Fourier Transform (QFT) and Quantum Phase Estimation (QPE) circuits as subroutines for factoring and period finding.

TLO-10: Simulate quantum system dynamics (Hamiltonian evolution) and solve linear systems using the HHL algorithm.

TLO-11: Train Parametrized Quantum Circuits (PQC) and Hybrid Neural Networks (QNN) using backpropagation and variational techniques.

TLO-12: Solve combinatorial optimization problems (Max-Cut, TSP) using the Quantum Approximate Optimization Algorithm (QAOA).

Let’s begin our quantum journey!