Introduction to Quantum Computing: Exam Revision Notes

By Petros Wallden, Raul Garcia-Patron and Joschka Roffe

Exam Format

- Date & Time: Tuesday 09 December 2025. Time 09:30-11:30
- Two questions with equal weight.
- Each question divided into two parts with each one multiple sub questions.
- *NOTES AND CALCULATORS PERMITTED examination: candidates may consult up to ONE A4 page (2 sides) of notes. CALCULATORS MAY BE USED IN THIS EXAMINATION*
- *Note: In the Exam we are evaluating the non-programming parts of the course (i.e. material covered by Raul, Petros and Joschka)*

Advanced Introduction to Quantum Computing (AIQC)

- Same as IQC in format and time of exam
- One question identical with IQC
- One question different with IQC (since both questions are from the “advanced” material parts)

General Advice

- **Read all questions before choosing and be "strategic"**
- **Questions** are divided into different parts:
  - Each part of a question is divided into small subquestions (weight is clearly stated).
  - In general subquestions hardness increase gradually, but not always.
  - Balance distribution of easy, medium and hard subquestions.
- **Do good time management!**
- **Exam with notes (2 sides) and calculator**
- **Past papers**: Only 2020-2024 IQC exams were open book(-ish).
  2015-2019 had more theory questions and some material that is no-longer covered in IQC.
- **Study Tutorials & train with past years exams** (check assignments)!
- We encourage you to use Piazza as a mutual help support tool, but we will not monitor Piazza ourselves next weeks.

The Basics of QC and QM (Raul’ part)

Disclaimer: These are general indicative guidelines (all material unless explicitly mentioned is examinable)

- Lecture 2,3,4,5 (Postulates): Material essential for next chapters. Need to know (for example): basic linear algebra and complex numbers, quantum states, operations and measurements, tensor products, inner-products.
- Lecture 6 (Circuit model): crucial to recognize most important gates and be able to compute the quantum state along the different layers of a quantum circuit and a final measurement outcome probability.
- Lecture 12 (Projective measurements): projectors and their use in measurement and Hadamard Test later.

Quantum Algorithms I (Raul' part)

- Lecture 7 (Phase Kickback and Deutsch-Jozsa): Deutsch-Jozsa, parallelization, phase kickback, Walsch-Hadamard transform, outcome probability resulting from quantum interference. Quantum/classical oracle of a Boolean function and phase kickback.
- Lecture 9 (Bernstein-Vazirani): Classical reversible circuit and its quantum analogue. Existence of quantum oracle from classical one. Building a phase kickback unitary from the quantum oracle. Bernstein-Vazirani algorithm.
- Lecture 10 (Grover): representation on 2-dim space, Grover iterations, phase kickback reflection and reflection over initial state. Estimation of number of iteration needed to reach solution.
- Lecture 13 (Simon's Algorithm): Projective measurement. Measurement of a subsystem. Understand the classical post-processing. Understand why the outcome of lower register is not necessary for solving the problem (just a trick to simplify the proof).

Quantum Algorithms II (Raul’ part)

- Lecture 15 (Parity Check) From the math definition of measurement to a circuit. Parity check of two qubits (crucial concept in quantum error-correction). Derivation of circuit. Parity on X basis.
- Lecture 16 (Hadamard Test) General Hadamard Test. Use of projectors to separate subspaces and define the measurement outcomes (probabilities and updated states). Two single qubit SWAP Test example. Two N-qubit generalization.
- Lecture 18 (Quantum Fourier Transform): Binary representation of integers and fractions. Fourier transform definition. Expression of output using binary faction notation. Iterative construction of QFT circuit. Counting of gates.
- Lecture 19 (Quantum Phase Estimation): Action of an exponentiation of U on an eigenvalue of U. Parallelization of the measurement register and control of different exponentiations of U. Connect the output with QFT to understand that the outcome of measurement is the eigenvalue of U. General input case. Order finding and factoring (subroutine of Shor’s algorithm).

Near-term QC (Petros' part)

- Lecture 21 (VQA I)

(i) Understand what VQA/VQE is. Motivation, mathematical prob., relevance and the four steps.

(ii) Understand Max-Cut to Ising Spin mapping. Mapping solution (cut) to spin configurations and think how this generalises.

- Lecture 22 (VQA II)

(i) Energy estimation. Including decomposing Hamiltonian to Pauli.

(ii) Ansatz states. Understand the concept of parametrised circuits/states. Be able to produce output state in simple examples. Compute expectation values of observables on output states.

(iii) Classical optimisation. Concept and gradient decent (see example in tutorial 6)

(See Tutorial 6)

MBQC (Petros' part)

- Lectures 23,24,25: Need to know

(i) basic MBQC patterns for single-qubit, two-qubit gates and how they are combined.

(ii) how to read an MBQC pattern to obtain the corresponding unitary implemented

(iii) how corrections work on angles (when measurement outcomes are not 0)

(iv) how to correct the output state (when measurement outcomes are not 0)

(v) how UBQC works (protocol and intuition). No details of the proof are required.

(See Tutorial 7)

QML (Petros' part)

- Lectures 26: Need to know

(i) General info on Supervised Learning (what it is, and how, inlc percetron and SVM in general, but not the details of SVMs)

(ii) The potential advantages (and obstacles) the QNN can have

(iii) QNN: encoding (be able to encode a bit string input to a quantum state, with the examples given in the lecture); training what cost function (cf VQAs for gradient decent); what is the output

(iv) Quantum Kernels: How and why to use Quantum Kernels. Be able to compute the Kernel (inner product) using quantum encoding given.

QEC (Joschka's part)

- For revision (and last tutorial): See Next Week (Thursday 27th November)

Tutorials & Coursework Specifics

- Tutorials: all.
- Assignment/CW: No coding question in the exam. Still could be useful to see the theory of the CW (but assignment usually harder).

Quantum Informatics CDT

THE UNIVERSITY of EDINBURGH

HERIOT WATT UNIVERSITY

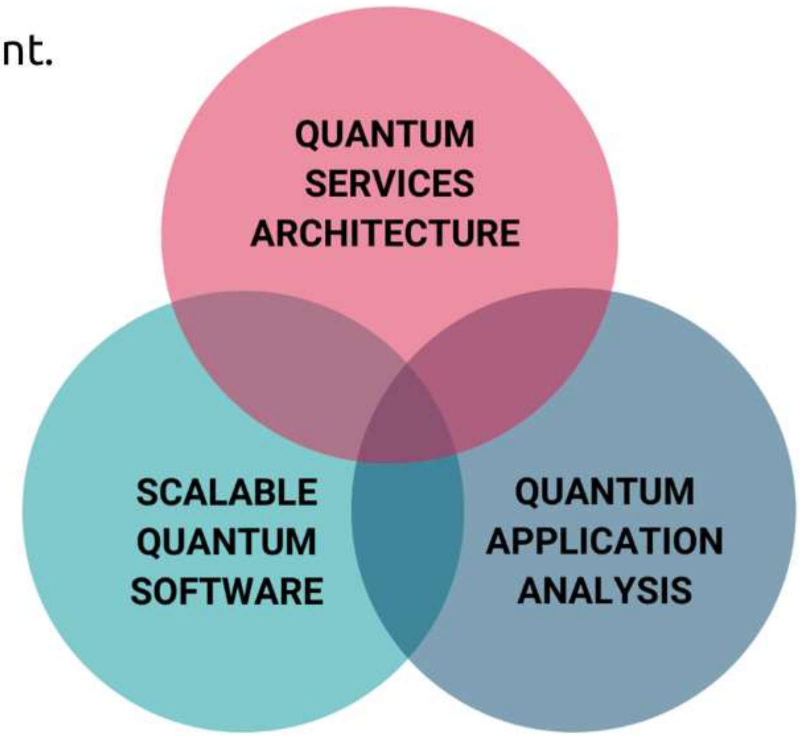
UCL

University of Strathclyde Glasgow

UNIVERSITY OF OXFORD

- 80 PhD studentships in 5 cohorts
- Fully funded 4 year programme
- Year 1 at Quantum Software Lab
- Years 2-4 at five leading Universities
- 70+ Academic experts
- 30+ Industry partners
- 10+ International universities
- Entrepreneurship programme
- Residential symposium
- Spring school

- Bridges computer science, mathematics, physics, and engineering.
- Make quantum technology interoperable and impactful.
- Interaction of quantum hardware, software, and applications.
- Social and ethical implications of innovation in quantum informatics.
- Unique collaboration of academia, industry, and government.



https://quantuminformatics-cdt.ac.uk