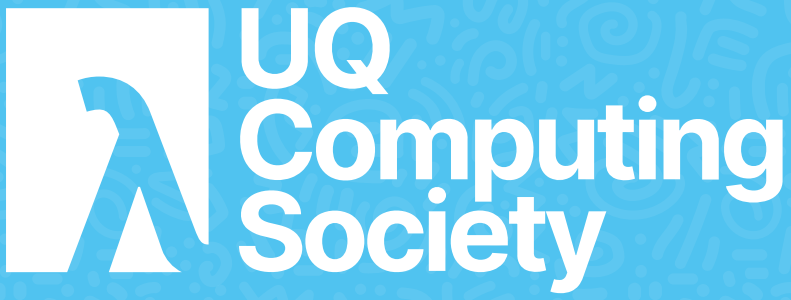
The University of Queensland Computing Society

FIRST YEAR GUIDE

2025 Edition

Learn. Code. Create.

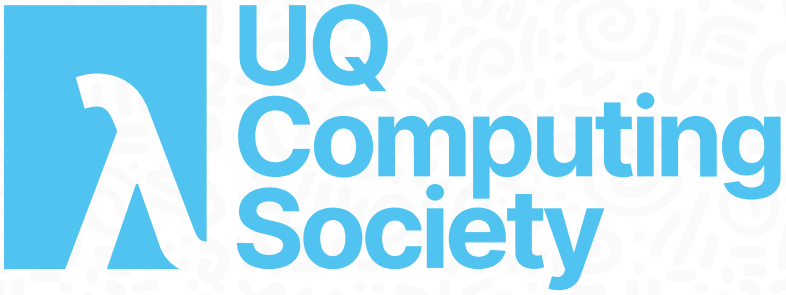
PUBLISHED BY



in 2025

Compiled by the 2025 UQCS Committee to guide your journey.

TABLE OF CONTENTS


UQ
Computing
Society

WHO WE ARE

The University of Queensland Computing Society (UQCS) brings together students with a broad range of interests ranging from digital design to software engineering. Our weekly events cover a variety of educational, social, and industry-oriented emphases to contribute to the development of the next generation of innovators.

In addition to our regular events, we host several major events throughout the year. Our flagship events are CodeJam (Semester 1), Hackathon (Semester 2), and High Frequency Networking (Semester 2). We also host trivia nights and industry events such as our Industry Panel and Mock Interviews in Semester 1.

Each year, we attract *over* 500 paying members and facilitate an extremely active Discord community. Our membership includes undergraduate and postgraduate students, as well as recent graduates and experienced alumni, some having been active in the society since our founding in 2011. In 2019, UQCS was awarded the **Best Faculty Club of the Year** from amongst UQ's 220+ clubs by the UQ Union, and in 2020 our regular tech talks were crowned the **Best Event Series** at UQ.

>500
MEMBERS

2200+
DISCORD MEMBERS

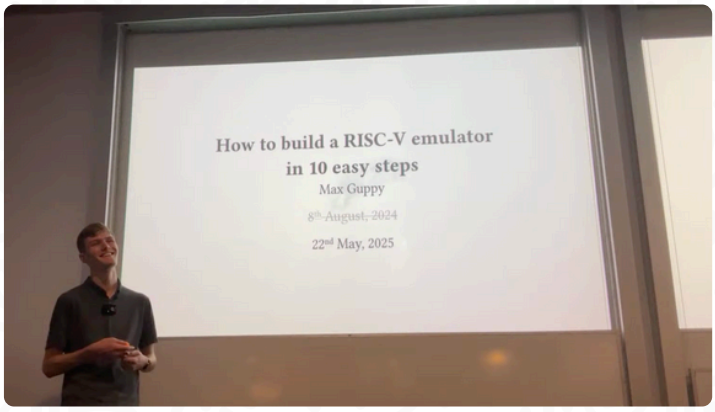
1000
PIZZAS

2100+
FACEBOOK FOLLOWERS

CodeJam, Semester 1 2025

WHAT WE OFFER

TECHNICAL TALKS



INDUSTRY AND NETWORKING EVENTS

CAREER BUILDING

COMPETITIVE PROGRAMMING GROUP

HACKATHONS

SOCIAL EVENTS

MEET THE COMMITTEE

The 2025 UQCS Executive Committee (T6) is the team leading the society this year. They organise events, manage partnerships, and keep things on track with the support of the general committee.

President

Richard Yu
(@jacron0085)

Secretary

Jesse Wang
(@jwjwjw99)

Treasurer

Zain Al-Saffi
(@nozer0)

Industry Officer

Masham Siddiqui
(@xmishi)

Events Officer

Tyreece Paul
(@tytross)

Media Officer

Naveed Rafi
(@zenopolous)

MEET THE COMMITTEE

The general committee helps with events and projects. They're a key part of keeping UQCS running!

Anshul Dadhwal
(@anshul.cpp)

Yuvraj Fowdar
(@antiraedus)

April Kidd
(@oatalicious)

David Long
(@_june71)

Chris Meng
(@cclownn)

Cielo Nicolosi
(@bananananananana192)

Harsha Varma
(@hash_v)

Blåhaj

SEMESTER 1 EVENTS

Week 0

Semester 1 Market Day
UQCS has a stall at every O-Week Market Day, alongside 220+ UQU clubs and societies. Come meet the team, grab some freebies, and find out what's happening this semester.

Week 0

Tech Clubs BBQ
The Tech Club BBQ is a welcome event hosted early in the semester by Brisbane's tech societies!

Week 1

First Year Panel
A casual Q&A where first-years hear directly from current students, graduate engineers, researchers, and a lecturer. Panelists share their experiences with internships, uni life, and how they got started in tech.

Week 3

Optiver Volatility Lab (UQCS x UQLIT x UQMSS)
A hands-on workshop run by Optiver where students explore market-making and trading through games, simulations, and real-world strategy.

Week 3

Jane Street: Get to Know Strategy & Product
An on-campus info session where students learn about Jane Street's approach to strategy and product roles, followed by Q&A.

Week 4

Resume Roasting
See what makes a great resume using real student submissions. Learn what works, what doesn't, and how to stand out to recruiters. Perfect for anyone updating or starting from scratch.

Week 5

Career Launchpad
An intro session on how to start preparing for internships and grad roles early in your degree. Covers key tips, timelines, and how UQCS can support your journey.

SEMESTER 1 EVENTS

Week 6

Cracking the Behavioural

Learn how to structure strong answers, speak with confidence, and stand out in software and trading interviews.

Week 7

Industry Panel

Hear from engineers and recruiters from Amazon, Swyftx, Oracle, VivCourt, and TechnologyOne. They'll share insights on careers in tech, landing roles, and what they look for in students.

Week 8

Mock Interviews (UQCS x UQLIT)

Practice 1-on-1 behavioural interviews and connect with industry reps from Swyftx, Deloitte Digital, Oracle, Canva, and more. Perfect prep for internship and grad application.

Week 9

CodeJam

CodeJam is UQCS's flagship coding competition where students solve programming challenges in teams. A prep session, How to CodeJam, is held beforehand to help first-timers get familiar with the format.

Week 12

How to Build a RISC-V Emulator in 10 Easy Steps

A deep dive into how emulators work by Max Guppy, a student who built one from scratch!

Week 12

Build for Impact Hackathon (UQCS x UQIES x UQU x EBESS)

A weekend-long hackathon focused on building tech projects with real-world impact, focusing on a solution for the UQ Union!.

Week 13

End of Semester Drinks

A relaxed social to wrap up the semester with drinks, snacks, and good company.

SEMESTER 2 EVENTS

UQCS reserves the rights to alter this events calendar based on society needs

Week 0

Semester 2 Market Day
UQCS has a stall at every O-Week Market Day, alongside 220+ UQU clubs and societies. Come meet the team, grab some freebies, and find out what's happening this semester.

Week 1

AllUni Competition
An inter-university ICPC-style contest with $5000 in prizes, open to all skill levels. Held at QUT Gardens Point with free food and sponsor merch.

Week 2

Semester 2 Launch Party
Kick off the semester with a joint social hosted by UQ's tech clubs. Meet new people, enjoy food and games, and find your community.

Week 3

Hackathon and How to Hackathon
A weekend-long coding comp with a $5000 prize pool, five fully catered meals, unlimited Red Bull, and exclusive merch. New to it all? How to Hackathon runs beforehand to help first-timers get prepped.

Week 4

SIG Brainteasers
A logic puzzle night hosted by SIG with UQCS and UQ Fintech featuring tricky brainteasers, food, and prizes. Great for anyone who loves thinking outside the box.

Week 4

Jane Street Estimatathon
A fast-paced team-based guessing game combining math, logic, and chaos — run by Jane Street with swag and food provided.

Week 5

GPU Architecture
A deep overview of what it is like to actually develop fast GPU code and the dangers of it by Shromm Gaind

Stay up to date with our events through our Discord, Instagram, and Facebook!

UQ Computing Society

SEMESTER 2 EVENTS

UQCS reserves the rights to alter this events calendar based on society needs

Week 6

CPU Architecture
A deep dive into the different cpu architectures and how to develop code for them by Matt Young

Week 7

High Frequency Networking Night
Our premier networking event, in collaboration with UQLIT and EBESS, offers top sponsors a direct line to leading recruiters, talented graduates, and experienced senior engineers. Connect and prepare for the upcoming internship season.

Week 9

Hiking
A hike along the mountains as we socialise and enjoy a nice BBQ on the way back.

Week 10

AGM
Join us for our Annual General Meeting (AGM), where we'll elect the committee for the upcoming year. This important event provides an opportunity to shape our future leadership.

Week 12

Study Night
In preparation for the final weeks, join us as we run sessions to help prepare you for either your final exams or assessments!

Week 13

End of Semester Drinks
A relaxed social to wrap up the semester with drinks, snacks, and good company.

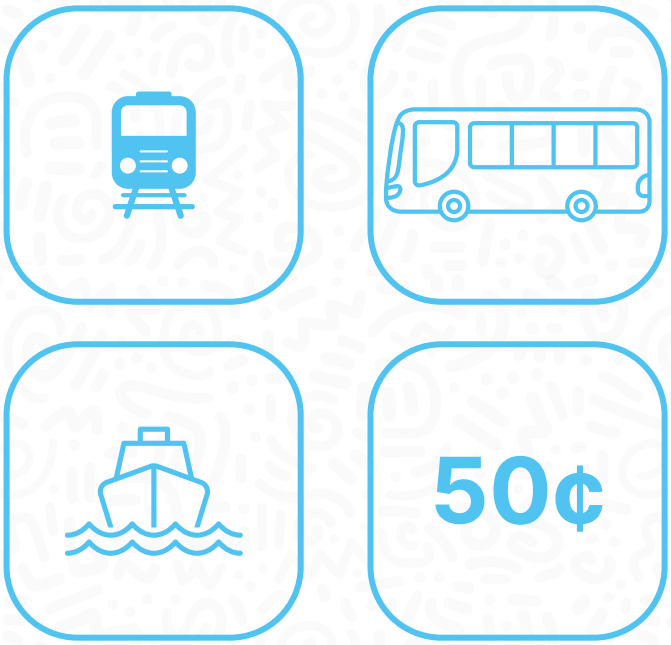
Stay up to date with our events through our Discord, Instagram, and Facebook!

UQ Computing Society

MUST KNOWS

for your first year!

GETTING TO CAMPUS



PUBLIC TRANSPORT

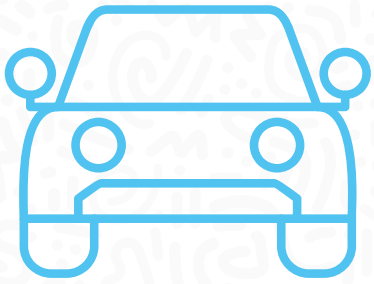
UQ St Lucia is well-connected by public transport, and your main options are bus, ferry, and (kind of) train. Use the **Translink Journey Planner** or **Google Maps** to find the best route based on where you're coming from.

There are two main bus hubs on campus:

- UQ Lakes – best for Southside and East Brisbane.
- Chancellors Place – best for North and West Brisbane.

Tip: If you are taking the M2 during peak time, it might be worthy to consider catching a train to/from Park Road Train Station instead!

DRIVING IN



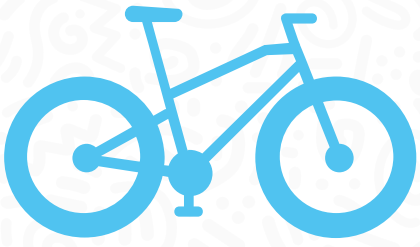
Depending on where you're coming from, driving part of the way can actually be faster. Many students choose to drive to a nearby busway and catch a quick bus to campus.

Parking is available on campus, but there are both capped and uncapped zones meaning some spots are much more affordable than others. You'll also need the CellOPark app to pay for parking so make sure this is ready before you drive in for the first time!

Always check the UQ Parking website for the latest rates and zone info.

Tip: Spots fill up by 9AM usually, so aim for earlier or later classes if you're driving in regularly. Also, parking is free on weekends and public holidays!

CYCLING IN



Cycling to UQ is quite common, especially if you live nearby. There are bike racks all across campus, and you can find them on **UQ Maps.**

If you want secure storage, UQ offers bike lockers in certain locations. Showers and change rooms are also available near the main bike facilities.

Find out more at
https://campuses.uq.edu.au/cycling/parking-facilities

HUNGRY ON CAMPUS?

UQ St Lucia has a variety of food and drink options on campus!

Chris Meng
(@cclownn)

> If you buy food, make sure you always get that **UQ mates rates.** But Fr tho the chips from **UQ main course** are soooo good.

Masham Siddiqui
(@xmishi)

> As someone who used to stay late on campus a lot and come in on weekends, **Hawken Village** on Hawken Drive was my holy grail. It is a short walk from Chancellor's Place and has HEAPS of food and drink options (even an IGA)! **Ciao Bello**, one our social sponsors, is a *delicious* Italian restaurant you can get a 10% discount at as a UQCS member!

HUNGRY ON CAMPUS?

Within the food, drink, and retail options at UQ, all UQU outlets offer Mates Rates discounts to UQ students!

Join UQU Mates Rates here https://uqu.com.au/food-retail-outlets/#mates-rates

Additionally, you can view outlet specific offers on their page found on the UQU website!

Check it out here https://uqu.com.au/food-retail-outlets/

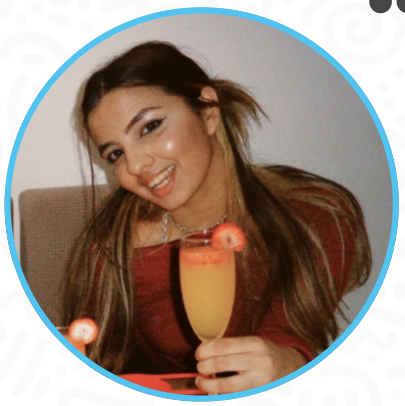
OUR PERSONAL ADVICE

Zain Al-Saffi
(@nozer0)

"The IGA near the campus sells 5 dollar packets of caffeine gum that dont make you crash if you need a quick boost to wake up.

University is what you make it to be, you can be make a comeback if you did not do well before and I have also seen smart people from high school really slip and bomb, don't fall into the trap of getting comfortable just because you cruised through high school. First year is all about experimenting and finding what factors influence your abilities the most (wether its emotional triggers to succeed, fun or external factors).

For the people that want to break into industry early, attend the hackathons, develop a support network you can rely on and work with and never skimp out on your fundementals. If you want an internship, it will never just magically come to you, you will have to show you want it more than the other applicants, develop your skills to stand out from your cohort."



Masham Siddiqui
(@xmishi)

"Success is non-linear, and your time at uni likely will not be either. Don't be afraid of change and analysing how a degree or study load fits your life. Experiment with different study loads, electives from other disciplines (if you have any), and genuinely reason with yourself if you are doing what will truly bring you happiness in the future.

If you have the time, do a summer semester (or two). Doing a summer semester can lower your study load during the semester and can get you ahead (plus pretty much all of them offer external mode!). Check out https://programs-courses.uq.edu.au/ for the full list."

Tyreece Paul
(@tytross)

"With AI tools like Copilot and LLMs becoming more popular for writing code, many news students are starting to rely on them heavily to the point where they can't code without them. These tools are powerful and useful for speeding up repetitive tasks, understanding new concepts or code reviews, but they should complement your skills, not replace them.

Overreliance on AI will only hurt the essential soft skills that make a great programmer: problem solving, creativity, discipline and debugging intuition. Focus on building a strong base: read the docs, write raw code, debug your errors, and struggle through the logic, make mistakes. That's how real understanding forms.

Read → Practice → Implement

The best way to learn a new topic is to build something with it. Start small with mini projects, questions, Leetcode, whatever sparts your interest! Implementation is what turns theory into practical skills."

OUR PERSONAL ADVICE

Chris Meng
(@cclownn)

> Make sure to have fun and enjoy uni so it's easy to motivate yourself to to attend. Because just showing up is already a great job <3

Yuvraj Fowdar
(@antiraedus)

> My single best experience in first year was attending the UQCS hackathon. But also, IMO the most important benefit of uni I found was the people you meet along the way, not the pure degree itself, so force yourself to do new things and talk to new people!! If you can, do courses that interest you AND/OR have good reviews. GYG is love/hate, kenko is consistently fine

Anshul Dadhwal
(@anshul.cpp)

> UQCS High Frequency Night and Mock Interviews are some of the best ways to network with industry experts, uqcs discord and secat scores are the best places to make informed decisions about the courses you'd want to take for your electives at uq (saves you a lot of tears and sleepless nights), never take two math courses in one semester unless you are a math major, smash out all the tougher courses in the first half of your degree so that you are only left with easy course in the second half to start gearing up for industry, neetcode and Abdul Bari on YouTube for getting your DSA basics right!

Brief

ACADEMIC ADVICE

for your first year!

DEGREE STRUCTURE

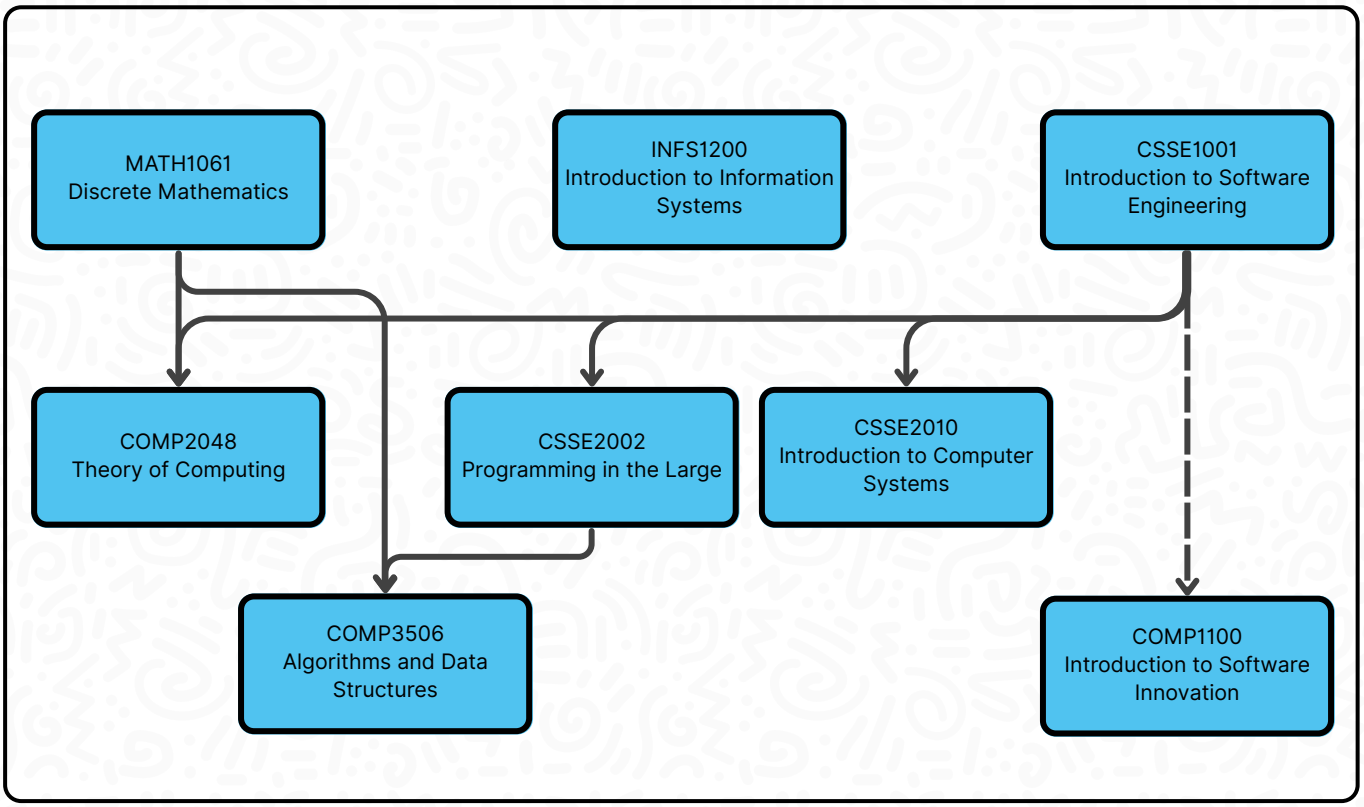
The following information is a general guide and may not reflect the specific structure of your program. UQCS strongly recommends regularly checking the official UQ program structure and course plans, as these can vary depending on your degree, major, and commencement year.

Bachelor of Computer Science (48 Units)

CORE	CORE	CORE	CORE	CORE	CORE	CORE	CORE
PLAN	PLAN	PLAN	PLAN	PLAN	PLAN	PLAN	PLAN
ELECTIVE	ELECTIVE	ELECTIVE	ELECTIVE	ELECTIVE	ELECTIVE	ELECTIVE	ELECTIVE

Core Courses Flowchart

Courses you need to do when completing a Bachelor of Computer Science (2025 Program)



- Lines mean prerequisites (e.g. CSSE1001 is prerequisite to CSSE2002)
- Dotted lines mean recommended prerequisites

FAQS

Are prerequisites enforced?

The following information is a general guide and may not reflect the specific structure of your program. UQCS strongly recommends regularly checking the official UQ program structure and course plans, as these can vary depending on your degree, major, and commencement year.

What happens if I fail a course?

Most UQ courses have hurdle requirements — mandatory assessments (like exams or projects) that must be passed to pass the course. If you fail a hurdle, you will fail the course even if your overall grade is above a 4.

Depending on the course and your performance, you might be eligible for a supplementary assessment or exam. This is a second chance to meet the hurdle requirement. If you pass the supplementary, you'll receive course credit, but the highest grade awarded is a 4 (Pass) and it will be written as a 3S4 on your transcript to indicate a supplementary was taken to pass the course.

If the course you failed is not compulsory, you may be able to replace it with a different elective. However, for mandatory courses, you will need to retake and pass them in order to meet your program requirements and graduate.

Do I need to choose a major?

No, selecting a major is not compulsory. Majors are structured specialisations designed by UQ to guide your study path. Though the single major BCompSc is already However, many students choose a "no major" pathway for more flexibility. We explore this option in detail in our Majors Overview section.

What if I'm in a double degree?

Students in a double degree usually take courses from both programs each semester -for example, two computer science courses and two from the other degree. Some degrees in EAIT (like Mathematics or Engineering) share overlapping courses, which can free up space for more electives in either program.

What if I've never coded before?

Many students begin the CS program without any prior coding experience. UQ's curriculum is designed with beginners in mind and starts from the basics. While some students may have prior experience, all students are taught from the same starting point. UQCS also provides additional resources for new coders, which you can find in our Resources section.

COURSES

Introduction to Software Engineering (CSSE1001)

What is the course?

This course serves as the first step into programming, using Python as the foundation. Designed for beginners, it introduces core programming concepts and problem-solving techniques essential to software development. You'll learn how to write clean, well-structured code and gain confidence in debugging and gain confidence in debugging and building simple software systems.

What will you learn?

- Fundamental programming concepts (variables, loops, conditionals, functions)
- How to debug and test your code effectively
- Basics of object-oriented programming (OOP) in Python
- Techniques for writing readable, maintainable and modular code
- Introduction to software development practices used by professionals

Course Highlights

- No prior coding experience required
- Hands-on Python exercises and weekly coding tutorials
- Build an application/game in assignment
- Early exposure to key software engineering ideas
- Builds strong foundation for computer science courses

Discrete Mathematics (MATH1061)

What is the course?

Discrete Mathematics is the mathematical foundation of computer science, introducing key structures and techniques used in logic, algorithms and computing theory. This course is designed to build the reasoning and abstraction skills essential for programming, algorithms, data structures and cryptography.

What will you learn?

- Prepositional and predicate logic, and methods of proof
- Sets, relations and functions
- Graph theory fundamentals and basic number theory
- Counting techniques and introductory probability
- Binary operations, groups and finite fields

Course Highlights

- Builds logical and mathematical foundations for computer science
- Strong focus on problem-solving and rigorous thinking
- Core preparation for algorithms, cryptography and theoretical CS

COURSES

Introduction to Information Systems (INFS1200)

What is the course?

Students will explore essential concepts in the world of data, including database design and implementation, relational database technology, data modeling and management, SQL querying, normalisation, and data security. This course in Information Systems is designed for beginners seeking hands-on experience and a solid introduction to working with data.

What will you learn?

- Understand fundamentals in data modelling and relational database design
- Learn to write and execute SQL queries to retrieve and manage data
- Apply normalization techniques to organise data efficiently
- Explore key concepts in data security and integrity
- Gain practical experience in designing and implementing databases

Course Highlights

- Gaining practical experience managing and querying real-world datasets using SQL
- Learning how to efficiently store, organize and optimise data for performance and accuracy
- Apply data modelling and normalisation techniques to ensure structured, reliable datasets

Introduction to Software Innovation (COMP1100)

What is the course?

This introductory course explores innovation through the lens of computer science and information technology. Students will work in teams on a discipline-specific project to learn how innovations are created and delivered. From ideation to prototyping, this course equips beginners with the tools, processes, and collaborative experience needed to participate in technology-driven innovation.

What will you learn?

- Understand the meaning and process of innovation in technology
- Learn how to work effectively in a technology project team
- Apply decision-making strategies within the innovation lifecycle
- Use prototyping tools and methods to bring ideas to life
- Develop skills to communicate and deliver innovative solutions

Course Highlights

- Hands-on experience in delivering a real-world innovation project
- Learning how to turn ideas into prototypes using industry-relevant tools
- Developing teamwork, communication, and project planning skills

COURSES

Programming in the Large (CSSE2002)

What is the course?

Programming in the Large builds on foundational coding skills, guiding students through advanced software engineering practices using Java. Through the construction of a complete application, students develop disciplined coding habits and learn to make thoughtful design decisions aligned with modern programming standards.

What will you learn?

- Apply object-oriented programming principles to structure and organize code effectively.
- Use unit testing, exception handling, and documentation to create reliable, maintainable software.
- Design and build scalable applications with a focus on software quality and best practices.

Course Highlights

- Build a complete Java application using professional coding techniques.
- Develop disciplined software habits through structured design and testing.
- Learn modern programming practices to write clean, efficient, and maintainable code.

Introduction to Computer Systems (CSSE2010)

What is the course?

Introduction to Computer Systems explores how computers work from the ground up, bridging the gap between hardware and software. Using both C and assembly language, students learn to build low-level applications while gaining hands-on experience with digital logic, computer architecture, and microcontroller-based systems.

What will you learn?

- Understand digital logic, circuits, and machine-level data representation.
- Explore computer organization, memory architecture, and system interfacing.
- Program microcontrollers and write assembly and C code for low-level system control.

Course Highlights

- Build physical applications by combining C, assembly, and hardware logic.
- Learn how computers execute code and manage data at the architectural level.
- Gain practical experience with low-level programming and building circuits.

COURSES

Theory of Computing (COMP2048)

What is the course?

Theory of Computing explores the foundational principles that define what computation is and what it can be. Through topics like Turing machines, lambda calculus, cellular automata, and quantum computing, students investigate the theoretical limits of computing and how these ideas shape the systems and technologies we use today.

What will you learn?

- Understand core models of computation, including Turing machines, finite state machines, and lambda calculus.
- Explore early cryptographic methods, cellular automata like the Game of Life, and the Church-Turing thesis.
- Gain exposure to emerging ideas in quantum computing and their implications for the future of computation.

Course Highlights

- Design and discuss three creative projects applying theoretical models to real-world or conceptual problems.
- Analyze the role of computation beyond modern hardware, including abstract and quantum systems.
- Develop a deeper understanding of the mathematical and philosophical foundations of computer science.

Algorithms and Data Structures (COMP3506)

What is the course?

This course introduces students to core data structures and algorithms, focusing on how abstract data types are represented and manipulated in memory and storage. You'll explore the foundations of efficient computation, including performance analysis and the use of key data structures like lists, trees, hash tables, and graphs. Mastery of these concepts is not only essential for building robust software but also critical for success in technical interviews and coding assessments.

What will you learn?

- Understand abstract data types and their real-world applications
- Analyse the time and space complexity of algorithms
- Implement core data structures
- Apply sorting and searching techniques to solve computational problems
- Develop problem-solving skills valued in software engineering and tech industry interviews

Course Highlights

- Developing efficient, scalable code using appropriate data structures
- Comparing performance and trade-offs of different data representations
- Building and manipulating data structures for use in real-world systems
- Gaining foundational skills that are heavily tested in technical interviews

MAJORS

Choosing a Major

At UQ, Computer Science students can choose from the following majors:

- **Cyber Security**
- **Data Science**
- **Machine Learning**
- **Programming Languages**
- **Scientific Computing**

Majors are specialised pathways that allow students to develop deeper expertise in a particular area of computer science. While all students must complete the core compulsory courses to graduate, majors offer a structured set of electives aligned with specific skill sets and career goals. Each major is explored in greater detail in its own section.
Bachelor of Computer Science students may complete up to two majors, allowing for a broader specialisation. However, students enrolled in a dual degree (e.g. Computer Science + Arts) may only complete one major within the Computer Science program. Regardless of the pathway, students are free to take electives from other majors, which can complement their focus and expand their capabilities beyond the formal title on their degree.

The No Major Pathway

The no major option is ideal for students who want flexibility and breadth in their degree. Rather than specialising in a single area, you can explore a wide range of topics and tailor your learning to suit your interests. Many students use this pathway to discover what they're truly passionate about. However, it's important to ensure your course choices still meet graduation requirements, so planning ahead is strongly recommended.

Students often suggest loosely basing your course selection around a major that interests you. This keeps your learning structured while still allowing room to branch out. Talk to peers, senior students, or members of the UQCS community to find out which courses are engaging or useful.

Most importantly, use this pathway to challenge yourself, not to avoid difficult courses. Take advantage of the freedom it offers to build a degree that reflects your interests, curiosity, and long-term goals.

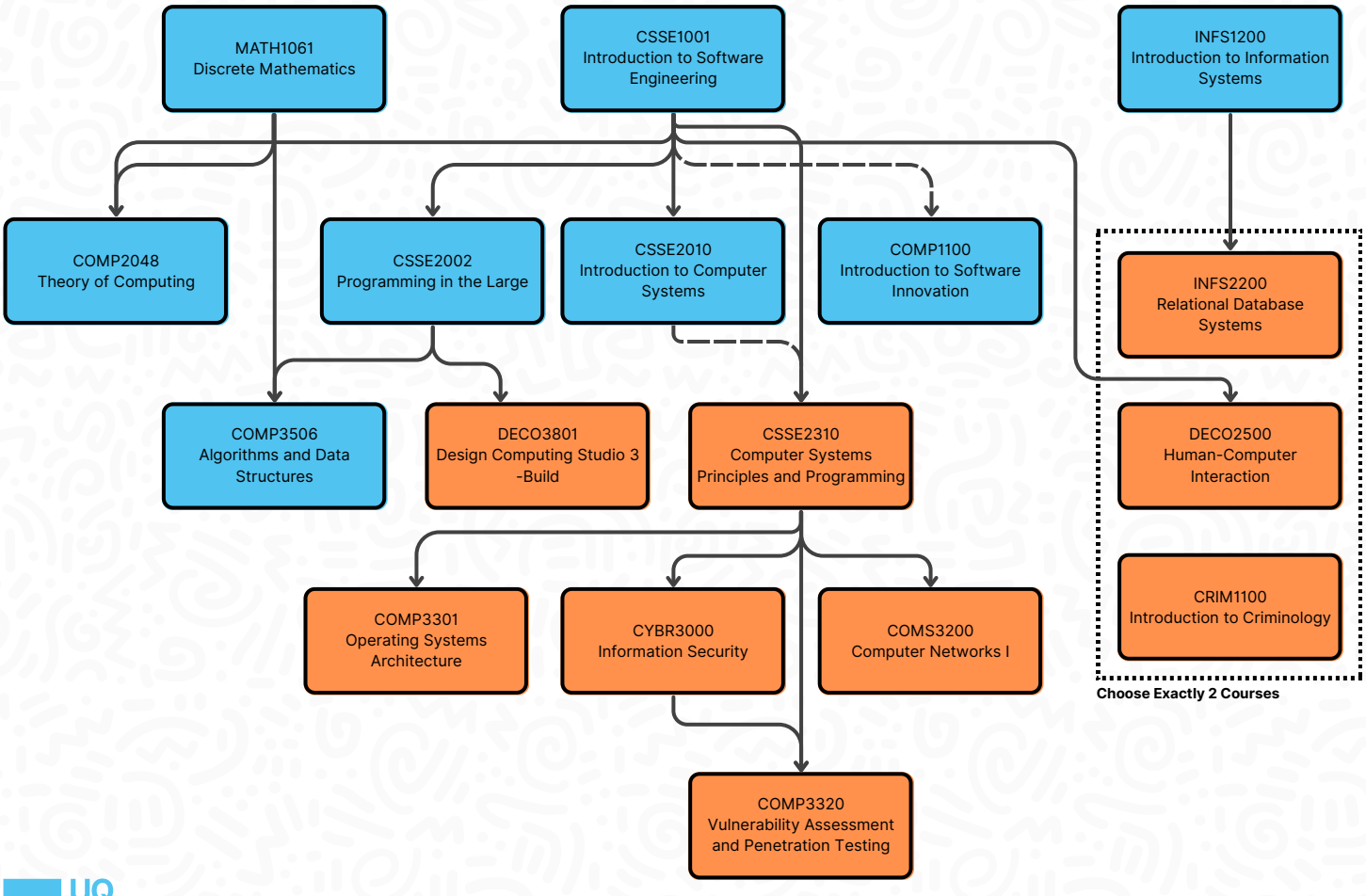
CYBER SECURITY

The Cyber Security major at UQ focuses on the theory and practical skills needed to protect digital systems and networks from malicious threats. Students will study core concepts in system and network security, secure programming, cryptography, and risk management. The program also includes hands-on experience in ethical hacking, penetration testing, and digital forensics, enabling students to understand how attacks occur and how to design effective defences.

Throughout the degree, students will explore the processes involved in identifying vulnerabilities, responding to security breaches, and implementing long-term security solutions. Emphasis is placed on both technical knowledge and the legal and ethical considerations of cyber operations. This prepares graduates to work across a range of domains where digital security is critical.

Career Possibilities:

- **Cyber security analyst**
- **Cyber systems engineer**
- **Security architect**
- **Information security officer**
- **Ethical hacker**
- **Software engineer**



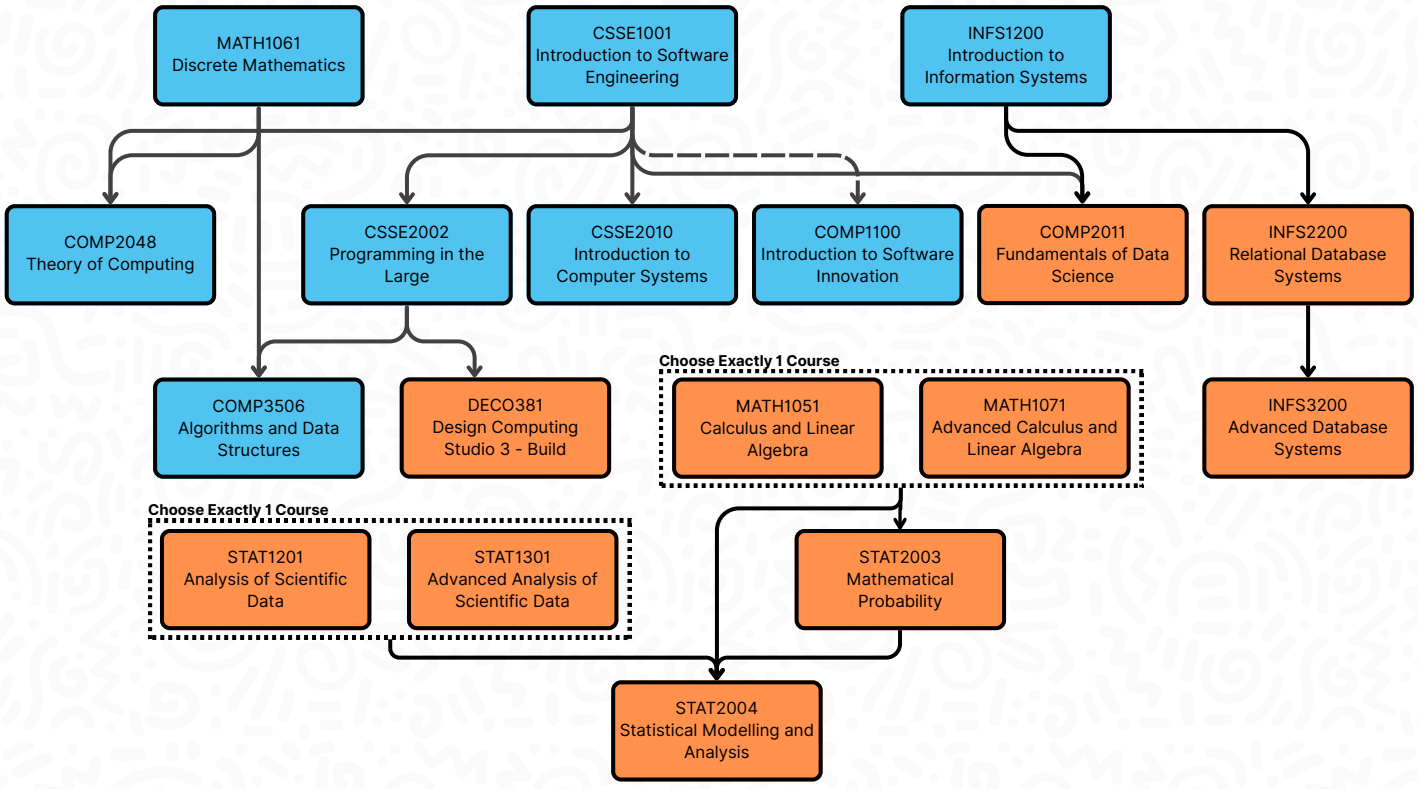
DATA SCIENCE

The Data Science major teaches you how to work with large-scale data, from collection through to analysis and interpretation. You'll study foundational techniques in data wrangling, statistical modelling, machine learning for data, and data visualisation, as well as tools for handling both structured and unstructured data. Courses also cover database systems, programming for data analysis, and the ethical considerations of data use.

This major prepares you to transform raw data into meaningful insights that support decision-making in business, science, and government. You'll gain the skills to work across the full data pipeline — from managing data storage to building predictive models — equipping you for roles that require both technical expertise and analytical thinking.

Career Possibilities:

- **Data Scientist**
- **Businesss Analysis**
- **Statistical analyst**
- **Database developer**
- **Research analyst**
- **Software engineer**



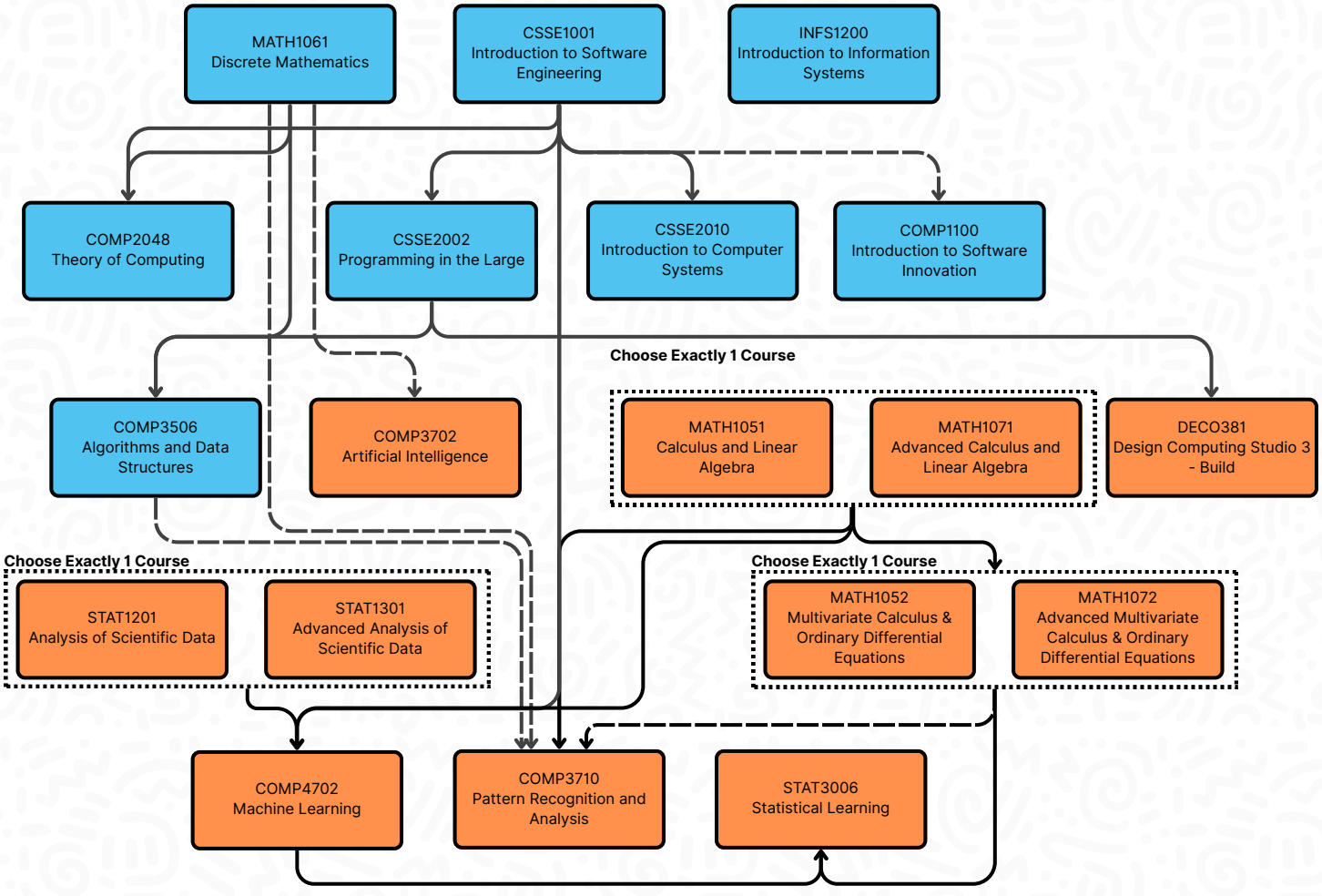
MACHINE LEARNING

The Machine Learning major focuses on the design and analysis of algorithms that enable computers to learn from data and make predictions or decisions without being explicitly programmed. You will study core topics such as supervised and unsupervised learning, neural networks, deep learning, probabilistic models and analysis. These are supported by foundations in mathematics, statistics and programming.

You will learn how to build models that can detect patterns, make forecasts and drive automated decision-making in complex environments. The major also explores real-world applications across fields like healthcare, finance, robotics and policy design, preparing you to contribute to the development of intelligent systems

Career Possibilities:

- **DevOps engineer**
- **Machine learning engineer**
- **AI Research scientist**
- **Software engineer**
- **Data scientist**
- **Research scientist**



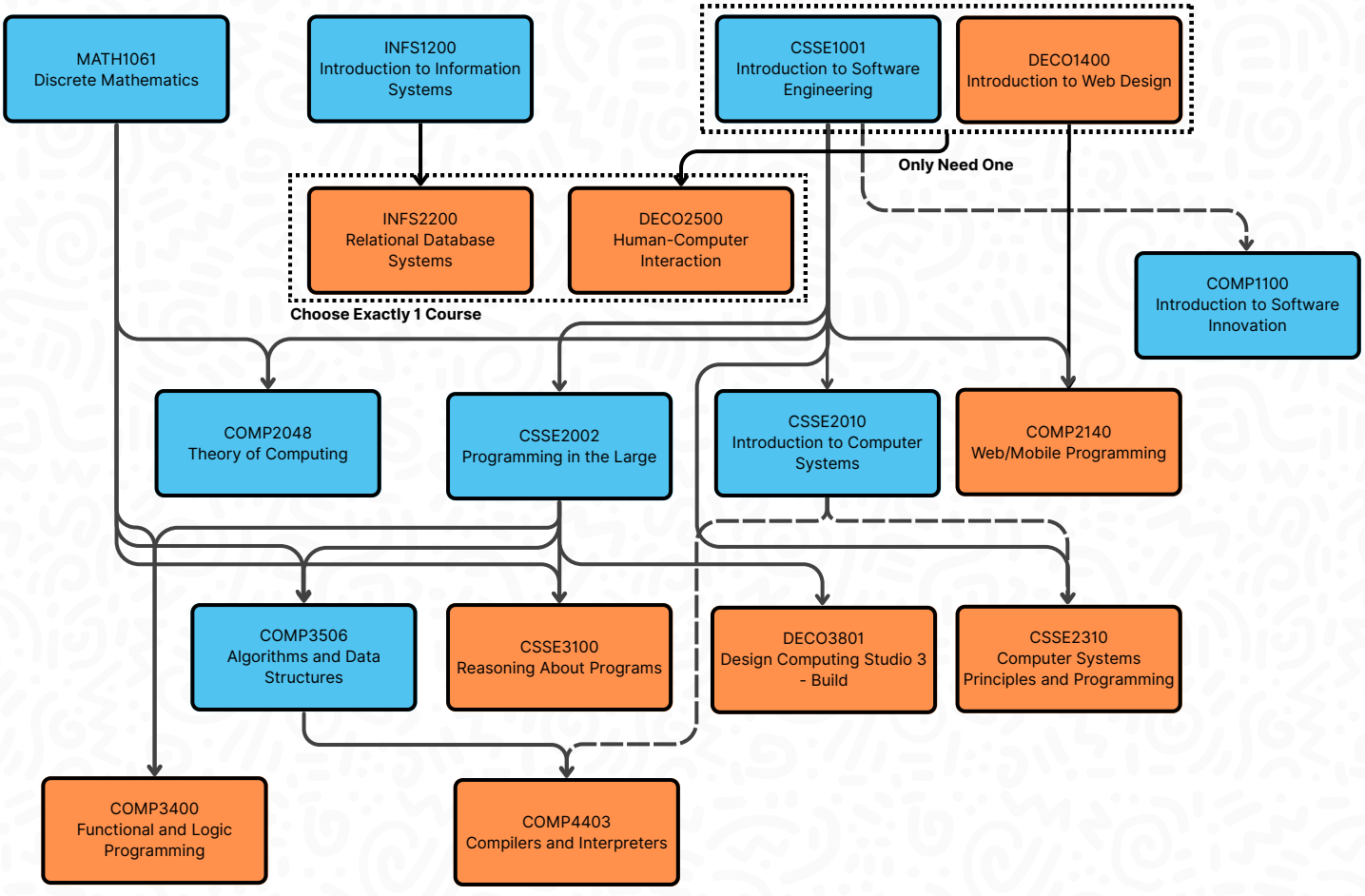
PROGRAMMING LANGUAGES

The Programming Languages major explores the principles behind the design, implementation, and use of programming languages. You'll study different programming paradigms—such as functional, object-oriented, and logic-based approaches—and learn how these influence the way software is written and executed. Core topics include language syntax and semantics, compiler construction, and software correctness.

This major focuses not only on writing code, but on understanding how languages work and how they can be improved or adapted for different problem domains. You'll gain deep insight into both the theory and practice of programming, supporting careers that involve building complex, reliable, and efficient software systems

Career Possibilities:

- **Software engineer**
- **Cloud engineer**
- **Web developer**
- **Application engineer**
- **Programming language developer**
- **Senior modeler**



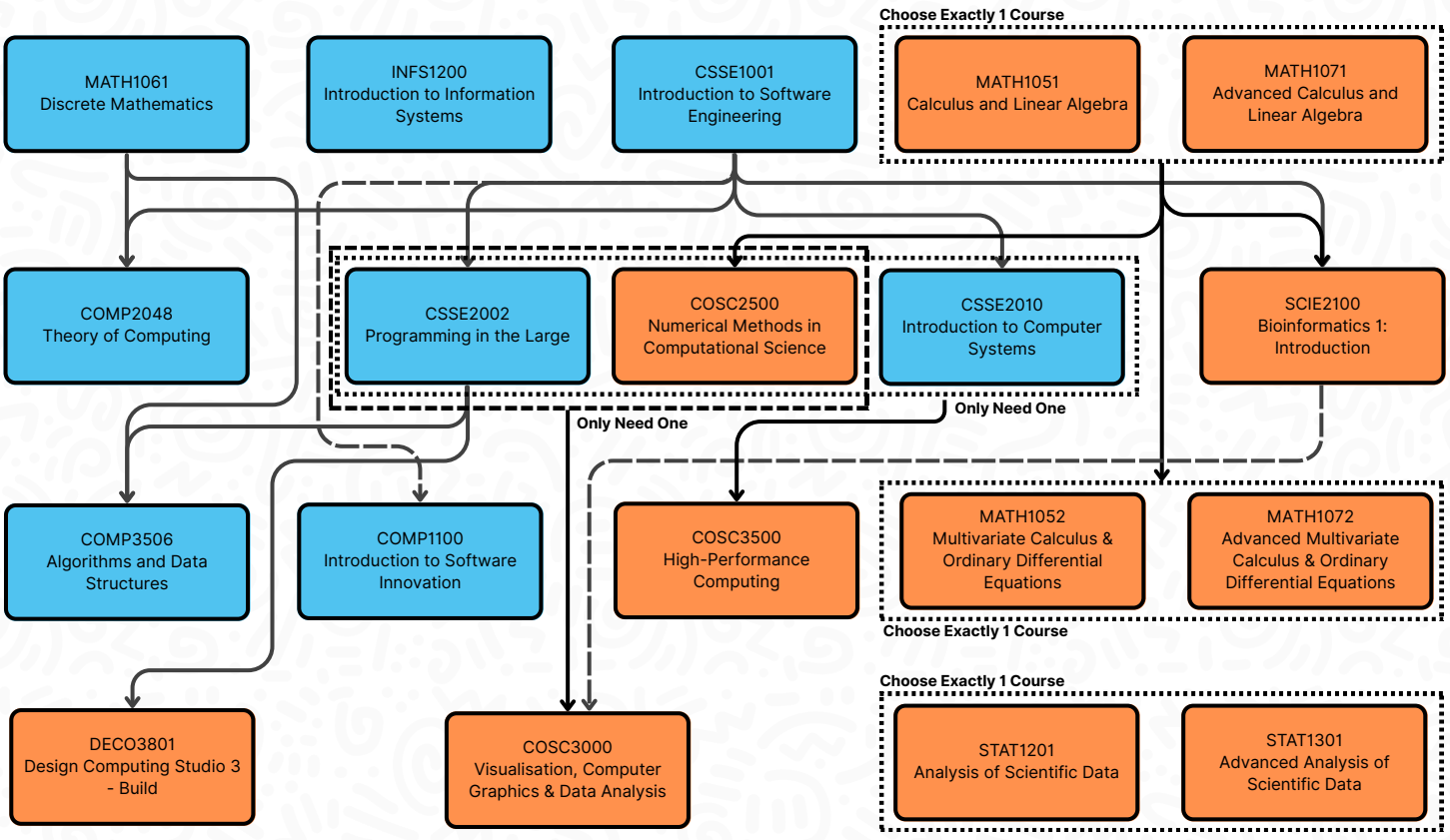
SCIENTIFIC COMPUTING

The Scientific Computing major focuses on the development and application of algorithms to solve complex mathematical problems using computers. You'll study how to translate continuous mathematical models into discrete forms that computers can process, learning numerical methods, computational modelling, and simulation techniques. This major blends applied mathematics, computer science, and algorithmic thinking.

You'll gain the skills needed to analyse real-world systems in areas such as physics, engineering, biology, and chemistry, using computational tools to make accurate predictions and support scientific discovery. These methods are widely used in hospitals, medical research, environmental modelling, and industries like pharmaceuticals and energy.

Career Possibilities:

- **Software engineer**
- **Technical business analyst**
- **Data engineer**
- **Computational Scientist**
- **Research Software Developer**
- **Algorithm specialist**



OUR SPONSORS

EMERALD TIER

Complimentary ACS student membership

Join ACS and launch your tech career **with personalised career support.**

Bridge the gap between your educational partner and work with the right skills and knowledge to enter Australia's crowded tech sector.

As an ACS member, you will:

- Receive personalised advice and support from your local ACS branch manager and team
- Understand the types of tech roles and career pathways available in Australia's ever-changing tech sector
- Gain relevant technical and vital interpersonal skills with unlimited access to ACS Skillsoft, a digital library of **35,000+ learning resources**
- Build contacts and relationships with employers with invitations to 400+ networking and professional development events a year
- Learn one-to-one with tech professionals prepared to share their expertise and **experience in the ACS mentoring program**

ACS is the professional association for Australia's tech sector and the largest community with 49,500+ members from across business, government and education. ACS is working with your educational partner to offer you a complimentary student membership.

Find out more at **acs.org.au**

How to get your complimentary ACS student membership

Join ACS today and gain access to ACS career advice and support.
Just follow the steps outlined below:

- Scan the QR code and visit the ACS membership web page. Click 'Join Now'
- Click 'Sign-up Now' and complete the email verification
- Select 'Supported Student ICT Membership'. The fee will be $0
- Select your educational institution from the drop-down menu
- Complete sections 1 and 2 with your personal and student details
- Click 'Browse Files' to upload your student ID or enrolment confirmation
- Click 'Submit' to activate your **complimentary ACS membership**

Scan me now

Should you have any questions, please contact ACS member services
member.services@acs.org.au

technology1

Making life simple for our community

Australia's Leading SaaS+ Enterprise Software Company

Technology1 (ASX:TNE) is Australia's leading enterprise software company and one of the ASX Top 100. Headquartered in Brisbane, they develop, sell, support, and run a fully integrated suite of enterprise software solutions that simplify complex processes for governments, universities, and businesses across the globe.

Their mission is to better communities — from citizens to students — by harnessing innovation to solve the complex and make life simple.

As a SaaS+ company, TechnologyOne delivers a powerful, secure cloud-first ERP platform (CiA) with built-in AI, GPS, and machine learning — accessible anywhere, on any device. With two major software releases annually, they continuously evolve to meet the needs of a digital world.

For students, TechnologyOne offers internships, graduate roles, and mentoring programs designed to build real-world skills in software engineering, design, product, and beyond.

Interested in joining a company building the future of enterprise software?

Check out opportunities:
https://www.technology1.com/company/life-at-techone/join-the-team

ORACLE

Powering the world's data and beyond

Oracle is one of the world's most influential technology companies. Known for inventing the modern relational database, Oracle has supported critical operations in finance, healthcare, telecommunications, government, and more for nearly five decades.

Today, Oracle leads with innovation across cloud infrastructure, data, and enterprise software. Oracle Cloud Infrastructure (OCI) delivers high-performance computing and secure storage, while the autonomous database simplifies management with self-securing, self-tuning capabilities.

For students starting their journey in tech, Oracle offers pathways through internships, graduate programs, and early-career opportunities designed to help you grow.
With a global presence and a mission to help the world run smarter, Oracle empowers organisations to solve complex challenges through intelligent, scalable, and secure technology.

Start your career at Oracle: oracle.com/careers

Jane Street

Work where your mind matters.

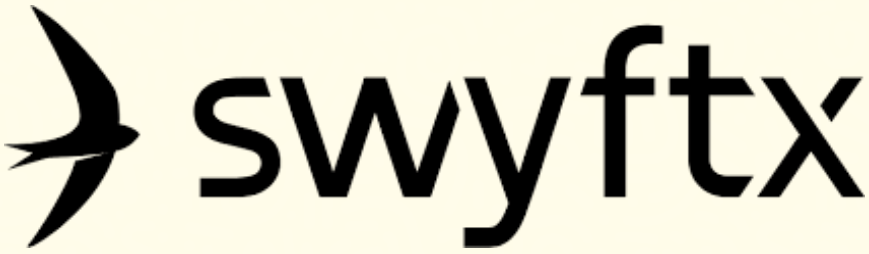
Apply to Jane Street today!

Jane Street is a global trading firm with offices in New York, London, Hong Kong, Singapore and Amsterdam. Our approach is rooted in technology and rigorous quantitative analysis, but our success is driven by our people.
We are always recruiting top candidates and we invest heavily in teaching and training.

- Check out our educational programs
- Take a look at our tech blog
- Solve some of our puzzles

Check out our jobs and internships:

Gold Tier



Silver Tier

SIG
SUSQUEHANNA

optiver

Bronze Tier

VIVCOURT TRADING

imc

CLOSING REMARKS

RESOURCES

UQCS

UQCS WEBSITE
TO LEARN MORE ABOUT UQ COMPUTING SOCIETY

UQCS DISCORD
OPEN TO ALL STUDENTS, ALUMNI, AND MEMBERS OF THE BROADER COMMUNITY INTERESTED IN DIGITAL DESIGN, COMPUTER SCIENCE, SOFTWARE ENGINEERING, IT AND MORE

UQCS INSTAGRAM
WHERE WE POST OUR SOCIAL MEDIA CONTENT AND CAN STAY UP TO DATE WITH THE CLUB

Useful Resources

UQATTIC
STUDENT CONTRIBUTED EXAM ANSWERS, USEFUL FOR EXAM PREPERATION ALTHOUGH USE WITH CAUTION AS THEY MAY NOT ALWAYS BE ACURATE

GITHUB
FOR EXTERNAL RESOURCES, CODE, PROJECTS AND MUCH MORE

LINKEDIN
FOR PROFESSIONALLY CONNECTIONS AND NETWORKING

KAGGLE
DATA SCIENCE AND MACHINE LEARNING COMMUNITY WITH COMPETITIONS, DATASETS AND LEARNING TOOLS

HACKERRANK
FOR COMPETETIVE PROGRAMMING COMPETITIONS AND RESOURCES FOR CODE THEORY

HACKTHEBOX
FOR CYBERSECURITY PRACTICE AND CODING RESOURCES

Interview Prep Tools

NEETCODE
FOR BEGINEERS LEARNING INTERVIEW STYLE QUESTIONS

LEETCODE
THE MAINSTREAM INTERVIEW PRACTICE AND INTERVIEW TOOL

GETCRACKED.IO
INTERVIEW QUESTIONS RANING FROM DESIGN PATTERNS, LANGUAGE KNOWLEDGE, OPERATING SYSTEMS AND MORE

PUBLISHED BY

UQ Computing Society

in 2025

Join UQCS: https://uqcs.uqcspay.org/
Join our Discord: https://discord.uqcs.org/

All Other Links Found at https://uqcs.org/
and linktr.ee/uqcomputingsociety