© The University of Queensland 2024
This content is protected and may not be shared, uploaded, or distributed.

The University Of Queensland
AUSTRALIA

COMP3301 Operating Systems Architecture

Dr. Matt D'Souza

Overview

- Operating Systems
- Course Requirements
- Timetable
- Assessment
- Course Outline
- Week 1 Lectures

Operating Systems

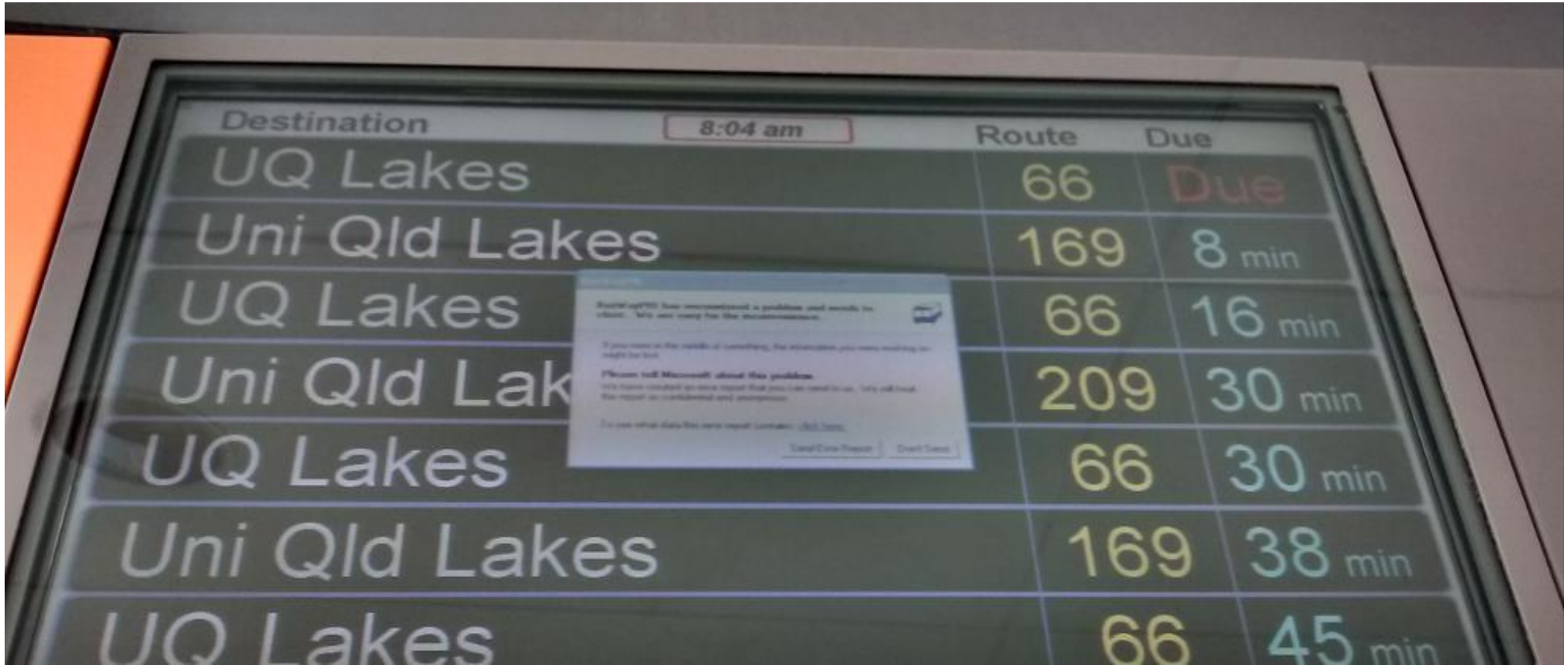
- The single most important category of software?
- The most interesting kind of software to work on?
- Most widely deployed software?
  - Tiny Embedded devices to tera-scale clusters
- Operating Systems are fundamental to computing!

Operating Systems

- How do they work?
- What are some of the (difficult) problems they solve?
- Why are they so important?
- What's it like to use and work on a real production OS?

Operating Systems

- OS are all around us! Even at bus stops!



Take this course if

- You want to *understand* computers, not just use them
- You can program well in C or are willing to learn C.
- You want to learn practical skills hacking on the world's most important OS
- To get experience with using large code bases.

Don't take this course if

- **You struggled with pre-requisite course**
- **You can't program well in C and can't spend time to learn C.**
- **You are are afraid of a bit of work**

Required C Knowledge

- **Basic C Knowledge Required (CSSE2310):**
- Pointers and functions
- If/else, for, while, switch statements
- bit manipulation: <<, >>, &, |, ^
- Shifting and masking
- Arithmetic functions: +,-,%
- structs
- strings and arrays
- Header files and defines: #include
- Commenting code style: /* */, //
- Use of macros: #ifndef
- Makefiles

Preliminaries

- **Coordinator/Lecturers**
  - Matt D'Souza – COMP3301@eecs.uq.edu.au
- **Tutors**
  - Sean, Lachlan ,Andrew, Yufeng, Jed and Harry
  - **DO NOT email/contact the tutors**
- **Post technical/assessment queries on Forum**
- **Course related questions – email Matt**
- **Always put COMP3301 in email subject and include your student number in the email**

Timetable

- **Lecture – Monday 12-2pm**
- **Contact - Tuesday 4-5pm (1 hour)**
- **Pracs Sessions (You can only attend you session)**
  - Tuesday 5-7pm, Wednesday 8-10am, Thursday 12-2pm, 4-6pm, Friday 8-10am –78-336
- **Sign-on to one two-hour session per week**
  - Starts in week 1
  - Course schedule is on BB
  - Changes will be given advanced notice

Learning Activities- Pracs and Quizzes

- **Practicals ("Pracs")**
  - Introduce key concepts used in the assignments.
  - Pracs are not used for assessment. They are used to help prepare you for the assignments.
  - Pracs are optional to do and are not assessed.
- **Weekly Online Quiz**
  - Weekly online Blackboard quiz, related to the week's lecture material
  - Quizzes are not used for assessment.
  - Provides self feedback on presented lecture material
  - Quizzes are optional to do and are not assessed.

Assessment - Assignments

THE UNIVERSITY OF QUEENSLAND
AUSTRALIA

- **Assignments must be submitted online (via code repo)**
- **Assignment 1 – 15% due week 5**
  - Process management
- **Assignment 2 – 25% due week 9**
  - Kernel programming
- **Assignment 3 – 25% due week 13**
  - Kernel Virtualisation or System Calls

Assessment - Exam

- **End of semester exam**
    - Marks: 35%
    - Two Hours
    - Closed Book

    - ***More details will be provided, later in semester***

Assessment – Pass Hurdles

- **To pass COMP3301 with grade 4:**
  - Exam – must get 40%
  - Assignment – must get 50% of the total marks assignments 1 to 3. (65%)
    - e.g a1 + a2 + a3 >= 32.5%

Extension/Late Assessment

- **See Electronic Course Profile (ECP)**
- **EXTENSION: You must apply to the EECS school office – see online form - to request extensions, medical certificate or similar required for extensions.**
    - Max 7 days from due date. Any longer and you should consider not submitting the assessment or dropping the course.
- **LATE: There is a 10% per 24 hours, late penalty for 7 days (EAIT faculty Rules).**

Student Access Plans (SAP) and Exam Adjustments (EA)

It is important you arrange your SAP or EA as soon as possible. Aim for the first four weeks of semester.

Who?

Students who:

- have a disability
- are neurodivergent (i.e. ADHD/Autism)
- have a mental health condition
- have a medical condition
- have an illness or injury
- have caring responsibilities

Or:

- are experiencing exceptional circumstances
- are an elite athlete
- have commitments to religious observance
- are engaged in defence service

What?

Student Access Plans:

- extension to assignments
- occasional absences
- alternative assessment formats
- placements, pracs, and lab adjustments
- accessible formatting
- assistive technology

Exam Adjustments:

- additional examination working time
- rest breaks
- use of a computer or a writer
- ergonomic furniture
- food or drink
- a separate room

How?

Meet with a Student Adviser

- visit Student Central
  - Building 42 St Lucia
  - Google: UQ Student Adviser
- contact Student Services
  - student.services@uq.edu.au
  - (07) 3365 1704

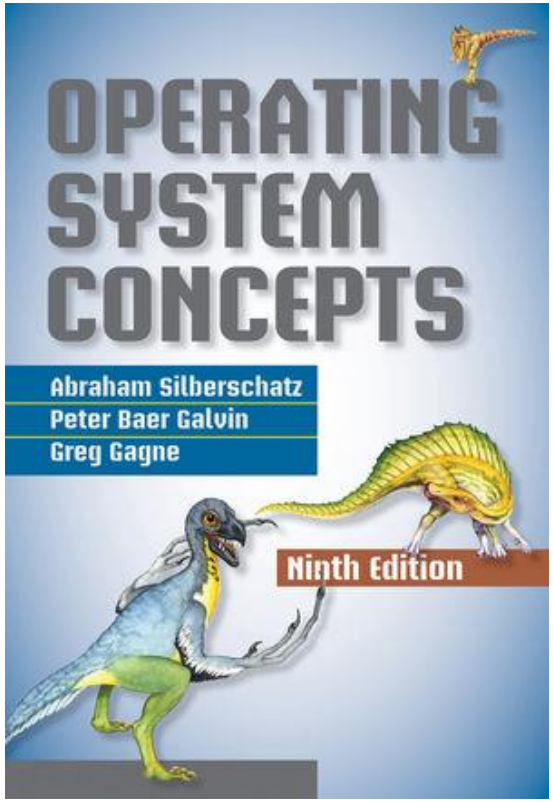
Get in touch as soon as possible if you need support for your study.

Support can be provided throughout semester.

Questions?

Textbook

- **Silberschatz, Galvin and Gagne, "Operating Systems Concepts", 12th Edition (or later), Wiley**



Assistance

- **Weekly Scheduled Prac sessions are your primary source of Assistance**

- If you have questions, then you should attend a prac session and talk to teaching staff in a one-on-one session.

- **COMP3301 ED Discussion forum site – secondary source of Assistance**

- Community help – anyone can respond with an answer.

- One-on-one help is not available on the ED Discussion board.

    - **Technical and assessment** related questions should be posted to the forum.

- To access the forum, go to BB and use the "ED Discussion" Link (left side purple menu).

- To save time, search first before posting as your question may have been answered.

Online Assistance

- **Any other course related matters (not technical related)**
    - Email: comp3301@eecs.uq.edu.au
    - **Please do not email any of the tutors.**
    - Use COMP3301 in the subject
    - Include student number in the email body.

Official School Online Course Resources

- **Online code repository**
- **Student email:**
  - All notices will be sent to only to your student email address
  - **It is your responsibility to ensure that you check your student email regularly for notices**
  - No concessions given for missed email notifications.

Private Tutoring

- **Private tutors are not endorsed or supported by COMP3301**
- **Private tutors are not permitted to use any of the code examples or COMP3301 equipment or facilities or the forum.**

Plagiarism/Collusion/Copying

- **Sharing of prac or assignment code is NOT permitted.**
- **Prac and assignment development is individual unless stated otherwise.**
  - **EECS takes collusion VERY seriously and will prosecute cases.**
    - Any cases of collusion or copying in exam, pracs and assignments will **NOT** be tolerated.
    - All submitted work (exam, pracs, assignments) **MUST BE YOUR OWN WORK**.
  - **Any suspicion of plagiarism/collusion WILL BE REPORTED to the EECS Academic Integrity Officer for further investigation.**

Plagiarism/Collusion/Copying

WE CHECK ALL OF YOUR CODE

./repo-master-3ca5f33404e1add928e1b5657567f3229fca5ee9/repo-master-3ca5f33404e1add928e1b5657567f3229fca5ee9/mylib/s4428426_cli_radio.c (47%)		./repo-master-daccf23406985871375973b3d9e29cc6a49a08ee/mylib/s435293 (40%)
267-298		777-808
208-225		383-404
482-490		614-625
310-320		487-499

```
/**
  * @brief  Fills the tx_buffer and give semaphore to the radio task to send.
  * @param  char cord[8] - string of 8 characters to fill the tx_buffer
  * @retval None
  */
void send_radio_position(char cord[8]) {
        //Fill payload of tx_buffer with XYZ
        char xyzWord[3] = {'X', 'Y', 'Z'};
        int temp, first, second;
        for(int i = 10; i < 16; ++i) {
                temp = s4428426_hal_hamming_encode(xyzWord[(i - 10)/2]);
                second = temp & 0xFF;
                first = temp & 0xFF00;
                first = first >> 8;
                tx_buffer[i] = first;
                ++i;
                tx_buffer[i] = second;
        }
        //Fill in xyz coordinates in tx_buffer to send via radio
        for(int i = 16; i < 32; ++i) {
                temp = s4428426_hal_hamming_encode(cord[(i - 16)/2]);
                second = temp & 0xFF;
                first = temp & 0xFF00;
                first = first >> 8;
                tx_buffer[i] = first;
                ++i;
                tx_buffer[i] = second;
        }
        myprintf("Sent from Radio: XYZ%s\r\n", cord);
        //Give the semaphore to the radio task to send the tx_buffer
        xSemaphoreGive(radioSem);
}

/**
  * @brief  Pen Command
  *             - Moves the plotter z position up or down depending
  *             - Sends a semaphore to the radio task
  * @param  writebuffer, writebuffer length and command strength
  * @retval None
  */

static BaseType_t prvPencommand(char *pcWriteBuffer, size_t xWriteBufferLen, const char *pcCommandString )

        long lParam_len;
        const char *cX;
```

```
/**
  * @brief  Set up XYZ packet to send to plotter
  * @param  coordinates[8]
  * @retval None
  */
void plotter_position(char coordinates[8]) {
        //Fill payload of packbuffer with XYZ
        char xyz[3] = {'X', 'Y', 'Z'};
        int temp, first, second;
        for(int i = 10; i < 16; ++i) {
                temp = s4352935_hal_hamming_encode(xyz[(i - 10)/2]);
                second = temp & 0xFF;
                first = temp & 0xFF00;
                first = first >> 8;
                packbuffer[i] = first;
                ++i;
                packbuffer[i] = second;
        }
        //Fill in xyz coordinates in packbuffer to send via radio
        for(int i = 16; i < 32; ++i) {
                temp = s4352935_hal_hamming_encode(coordinates[(i - 16)/2]);
                second = temp & 0xFF;
                first = temp & 0xFF00;
                first = first >> 8;
                packbuffer[i] = first;
                ++i;
                packbuffer[i] = second;
        }
        myprintf("Sent from Radio: XYZ%s\r\n", coordinates);
        //Give the semaphore to the radio task to send the tx_buffer
        xSemaphoreGive(radioSemaphore);
}

/**
  * @brief  Convert coordinates such that they can be read by plotter
  * @param  pos
  * @retval None
  */
void convert_position(int pos) {
        char xCoord[3], yCoord[3], zCoord[2];
        /* Set plotter's current position */
        /* Convert to ascii */
        itoa(instructions[pos][0], xCoord, 10);
        itoa(instructions[pos][1], yCoord, 10);
        itoa(instructions[pos][2], zCoord, 10);
        /* Check x coordinates */
```

Plagiarism/Collusion/Copying

What do Professional Freelancers think?

"Dear Dr. Matthew,

I'm a software engineer freelancer and I stumble upon job offers which are clearly an attempt to cheat at your course:

1) https://www.upwork.com/jobs/~01601d9386f57ef3ae (without an account you probably can't see attached files but this is how I found contact to you. I will attach them here as well)

2) https://www.upwork.com/jobs/~0107cead4eac9cc3c6 (which match description from pdf files)

I think this is despicable and I thought I should let you know."

The University Of Queensland
AUSTRALIA

Australian Government
Tertiary Education Quality and Standards Agency

CHEATING is NEVER the RIGHT ANSWER

TEQSA

Commercial cheating services are illegal in Australia

If you use cheating services, you could:

fail your subject or course

lose your visa

lose your professional accreditation

be blackmailed by cheating service operators.

Visit teqsa.gov.au/cheating for more information

! Remember: If you are having trouble with your assignments, contact your lecturer or tutor for help.

The University Of Queensland
AUSTRALIA

CHEATING is NEVER the RIGHT ANSWER

TEQSA

Australian Government
Tertiary Education Quality and Standards Agency

Commercial cheating services are illegal in Australia

It is illegal to sell or promote commercial academic cheating services

Criminal penalties include up to 2 years in prison and fines of up to $110,000

Students who use these services face disciplinary action in accordance with this institution's policies

Visit teqsa.gov.au/cheating for more information

! Remember: If you are having trouble with your assignments, contact your lecturer or tutor for help.

Student Services - Study Support

Learning Advisors- provide free one-on-one advice on assignment writing (breaking down tasks, starting assignment, understanding tasks- not proof reading), exam preparation/exam strategies, delegating appropriate time to each course/assessment, note taking, time and study management. Useful also if you are struggling with motivation and sticking to a routine.

https://www.uq.edu.au/student-services/learning

Learning Advisers (may be seen as "Learning and Writing Skills Development" in Studenthub).

https://www.uq.edu.au/student-services/learning

You are looking to book with any of the following:

Rosemary Graham

David Rowland

Kevan Jones Eva King

Michael Welch

Questions?

Course outline

- Processes and threads
- CPU Scheduling
- Deadlock and synchronisation
- Memory management and virtual memory
- Mass Storage
- File Systems
- I/O sub-systems
- Virtualisation, Protection and Security

Key Learning Objectives: What do you know about operating systems?

- **What is an operating system?**
- **What does it do?**
- **Why is it required?**
- **What sort of structures exist?**
- **How do users interact with an OS?**
- **How do programmers interact with the OS?**

Questions?

Development Environment

- Open BSD Virtual Machine
- All programming is done in C
- Coding IDE is up to you:
  - Code with an IDE like atom, sublime, nano, vim, etc
- Your code must conform to the course style guide provided

Code Repository

- **Your repo account will be issued by week 2.**
- **Your prac and assignment code must uploaded to your repo folder.**
- **Failure to upload code to your repository will result in failure of the prac or assignment.**
- **Your code will be checked for style correctness and for collusion.**

Questions?

Contact

Dr. Matt D'Souza
Lecturer

CRICOS 00025B • TEQSA PRV12080

© The University of Queensland 2024
This content is protected and may not be shared, uploaded, or distributed.