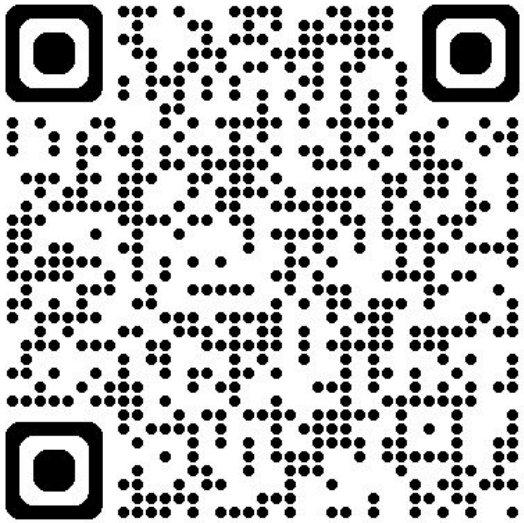
COMP1511/1911 Programming Fundamentals

Week 9 Lecture 2

Exam Information and Revision

Link to Week 9 Live Lecture Code

https://cgi.cse.unsw.edu.au/~cs1511/25T3/code/week_9/



My Experience Surveys



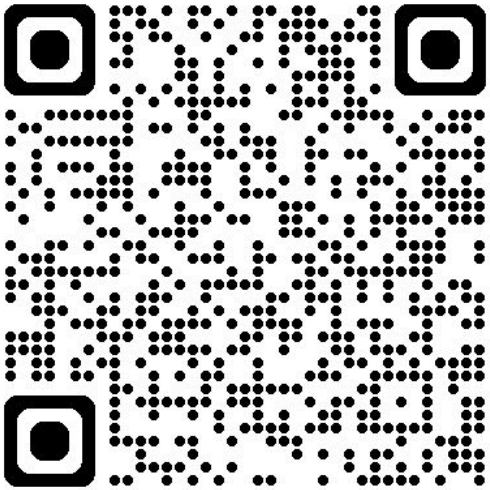
http://myexperience.unsw.edu.au/

Announcements

- **Assignment 1 Marks**:
  - Out now! Look at feedback not just mark
  - Message MARKER if you have any questions - zid is at the bottom
- **Revision Sessions:**
  - Last set of revision sessions on in week 11
  - Look out for announcement and sign ups on the forum soon
- **Drop in Sessions**
  - Email cs1511@unsw.edu.au to book a drop in session.

Week 10 Practice Exams

- Held in In Person Labs
- This is how you get
  - **Check off** and **Lab marks** for week 10!
  - Marks are given for
    - submitting at least 1 question.
- If you are in an online tut-lab
  - Sign up for an in-person lab for week 10 https://buytickets.at/comp1511unsw/2401109
  - You should still attend your regular tlb too.



Last Lecture

- Search and Delete
- Bigger Linked List Example - Email Management System

Today’s Lecture

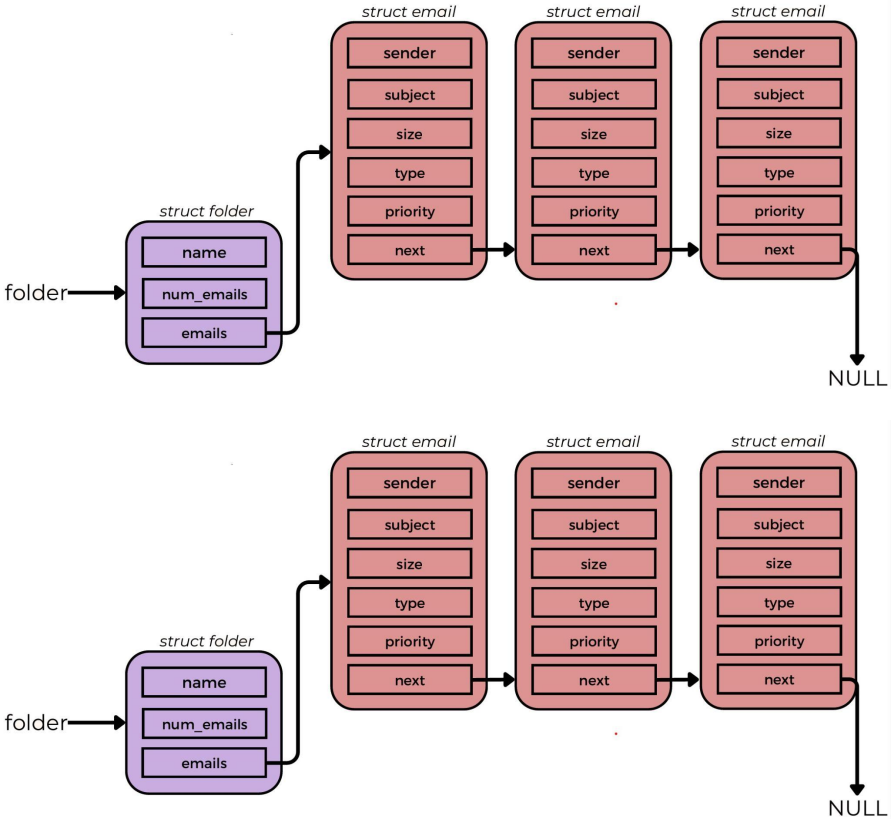
- Finish Email Management System last function: split
- Exam info
- Format
- Preparation
- Hints and tip
- Revision

Email Management System

1. Where was my memory leak for merge?
2. Lets implement split

Visualisation of the system

We can create many folders, each containing linked lists of emails.



Exam: What is in it?

- Everything that we have learnt so far
- Lots of focus on:
  - Variables: int, double, char, structs
  - Simple IF statements and WHILE loops
  - **Arrays, 2D Arrays**
  - Strings
  - Pointers
  - **Linked Lists**

The Exam: Time and Date and Location

- Date: Monday 30th November
- 3 hours + 10 minutes reading time
- There will be 2 sessions of the exam
  - **Wed 30/11/2026 Morning** (09:45-13:00)
  - **Wed 30/11/2026 Afternoon** (12:50 corralling 13:25 - 16:40)

Morning vs Afternoon Sessions

- Students in a **Morning** session
  - **are not permitted to leave the exam early**.
  - **who arrive late *may* be allowed to enter the exam** (up to 30 minutes).
- Students in an **Afternoon** session
  - **must be corralled** in a tutorial/lecture room before their exam.
  - **who arrive late are not permitted to enter the exam**.
  - **who arrive late to the corralling room are late for the exam**.
  - **may leave the exam early** but may not re-enter after doing so.

The Exam: Time and Date and Location

- There will be a form to pick your preference for your exam time later this week.
- Get in early to for the best chance of getting your preference.
- Your final confirmed times and locations will be sent to you by the exams team
- If you have clashes the exams team will put you in the appropriate time-slot

The Exam Environment

- The practice exam in the lab NEXT week and from week 8 lab is similar to your exam - so please familiarise yourself with the setup
- dcc, autotests, submit (like give) are the same as in your labs
  - Submit as many times as you want - only last submission will be marked
- **There will not be**
  - `dcc-help`
  - `dcc-sidekick`
  - `autotest-help`

The Exam: Exam Conditions

- Closed book, however:
  - Our course website will be open, which means you will have access and are allowed to refer to:
    - Any lecture material or code (no lecture recordings)
    - Any tutorial material or code
    - Any laboratory material or code

The Exam: Exam Conditions

- Exam conditions apply!!!
- If you experience any issues during the exam, please raise your hand and wait for an invigilator.
- No discussion of the exam or sharing your code with anyone except for COMP1511/1911 staff
- Do not communicate with anyone about the exam within 24 hours of the exam start time - this is considered plagiarism

The Exam: Exam Conditions

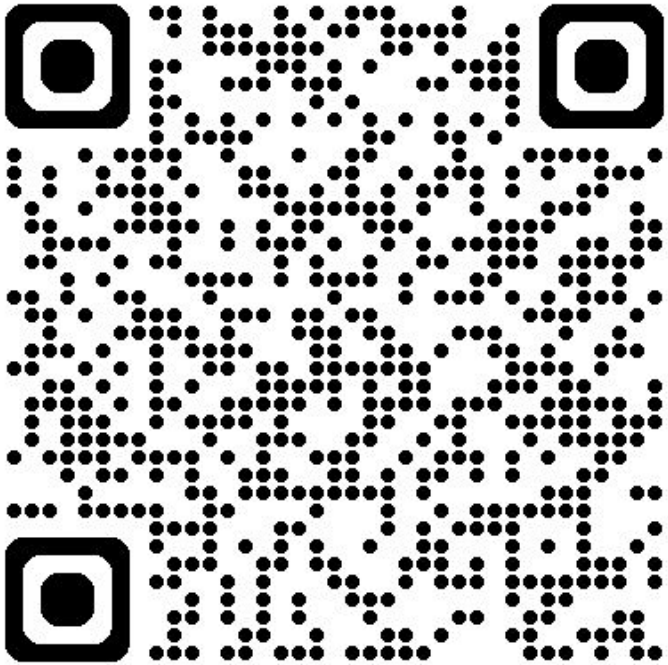
- You can bring a clear bottle of water
- Pens
- NO devices/phones etc
- No paper - we will give you a sheet of paper to do working on
  - This will be collected but not marked.
  - You can ask for more paper if you need it

The Exam: Exam Conditions

- When you come into the room and seat yourself, there will be instructions provided to you on how to start the exam
- You will have all the code in the home directory when you log into the exam.
  - This will be the same as in the practice exam
  - You do not need to fetch the exam.
  - You can re-fetch particular question files if needed.

The Exam: Official Exam Rules: Fit to Sit

- Fit to Sit Policy:
- By sitting the exam on the scheduled assessment date, you are declaring that you are fit to do so and cannot later apply for Special Consideration.



https://www.student.unsw.edu.au/exam-rules

The Exam: Feeling unwell during your exam

- If, during the exam you feel unwell to the point that you cannot continue with the exam
  - Stop working on the exam and immediately raise your hand.
  - Let the invigilator know you are unwell.
  - Submit your work and be escorted from the exam room to discuss with the course convenor/lecturer/admin
  - Obtain a medical certificate within 24 hours of the exam date
  - Submit a Special Consideration application within 3 working days saying that you felt ill during the exam and were unable to continue.

The Supplementary exam

- Tentative Date: Monday 11 January to Friday 15 January 2027
- Only for students granted special consideration
  - Applied for special consideration due to illness or misadventure
- If you think you will need to sit this exam, make sure you are available
- https://www.student.unsw.edu.au/exam-rules

The Exam: Hurdles

- There's an **array** hurdle, question **2 and 4**
  - You must earn a mark of 50% or more in at least one array hurdle question
- There's a **linked list** hurdle, question **1 and 3**
  - COMP1511 students must also earn a mark of 50% or more in at least one linked list hurdle question
  - COMP1911 students will still do these questions but they will not be hurdles
- These questions will be clearly marked on the exam paper as hurdles

The Exam Format

- 11 Practical Programming Questions
  - Rated (with 1 dot, 2 dot and 3 dots) to give you an idea of how difficult each question is
  - We hope that everyone can attempt and complete the first eight questions
  - We hope many students can attempt and complete q9 too
  - Not everyone will finish all questions in the exam! This is ok!!! You can still pass and get a good mark without finishing all 11 questions!!!

The Exam Format

- Q1 - Q4 (12 marks each) HURDLES
- Q5 - Q8 (5 marks each) Debugging Questions
  - These questions will be about whether you understand core coding concepts and the C programming language
  - In the style: "Debug the following code to make sure it works to produce the desired output."
- Q9 - Q10 (11 marks each)
- Q11 (10 marks)

Questions

- Questions are similar in style to the revision exercises and problem sets
- Rated (with 1 dot, 2 dot and 3 dots) to give you an idea of how difficult each question is.
- Some will have provided code as frameworks
- Each question will need to be written, compiled and tested
- You will have access to autotests
    - Harder questions will have less autotests

Questions 1 and 2: These are (●○○)

- Similar in style to Practice test questions 1 or 2
- Question 1 is a linked list hurdle (not a hurdle for COMP1911)
  - Use of linked list of ints/chars/doubles
  - No insertion or removal of nodes
- Question 2 is an array hurdle
  - Use of arrays of int/double/struct
- Tests your ability to:
  - Create simple C programs, use variables (int, double, char, structs) , use scanf and printf
  - Use **if** statements and **while** loops

Example Question 1 (●○○)

Perform some computation or comparisons on a linked list

Given a linked list, return the largest odd value in that list.
If the list is empty or contains no odd numbers, return -1
You can assume the list contains only ints >= 0
Edit the function

```
int largest_odd(struct node *head);
```

Example Question 1 (●○○)

Perform some computation or comparisons on 2 linked lists

Given 2 linked lists return the difference in the number of nodes in each list.

Edit the function

```
int difference(struct node *head1,
               struct node *head2);
```

Example Question 2 (●○○)

Loop through an array and gather some kind of information

Given an array:
Count the number of multiples of 3 in the array.

```
int array_count_multiples (int size,
                           int array[]);
```

Example Question 2 (●○○)

Loop through an array and gather some kind of information

Given an array of structs, where each struct is:

```
struct direction {
    int number;
    char dir;
};
```

Print out the total of the number of steps taken in a specific direction.

Example Question 2 (●○○)

So for example, if direction is 'l', find all the structs with direction as 'l' and add the numbers in those structs up.

Edit the function:

```
int step_count(
    int size,
    struct direction array[]
);
```

Questions 3 and 4: These are (●●◌)

- Similar in style to Practice test questions 3 and 4
- Question 3 is a **linked list hurdle** (not a hurdle for COMP1911)
  - You need to pass this OR Question 1 to pass the linked list hurdle
- Question 4 is an **array hurdle**
  - You need to pass this OR Question 2 to pass the array hurdle
- These are harder applications of the hurdles
- You may need to loop through more than once and/or insert/remove nodes, working with 2d arrays, test more difficult conditions and keep track of more than one thing.

Linked List Hurdle Example (●●◌)

`linked_list_duplicate.c`

Duplicate every node in the list by inserting the same node after the original node.

```
struct node *duplicate(struct node *head);
```

Linked List Hurdle Example (●●◌)

`linked_list_delete_duplicate.c`

Delete the first instance of a duplicate in the given list e.g.

1->3->5->3->1->7->X would give
1->3->5->3->7->X

```
struct node *delete_duplicate(struct node *head);
```

Array Hurdle Example (●●◌)

`array_multiplied.c`

Write a C program that reads integers into an array from terminal until a number is entered which when multiplied by another number previously entered results in 56. E.g.

```
./array_multiplied
2
3
28
28 * 2 = 56
```

Array Hurdle Example (●●◌)

`array2d_min_row_sum.c`

Write a C program that finds the sum of the minimum numbers in each row in a 2D array.

```
int min_row_sum(int size,
                int array[NUM_ROW][NUM_COL]);
```

Questions 9 and 10: These are (●●●)

- Question 9
  - Harder manipulation of arrays
  - Possibly fgets or string manipulation
- Question 10
  - Potentially use malloc() and free() with structs and pointers and/or command line arguments

More complex combinations, and some questions requiring interesting problem solving

Question 11: (●●●) and ☠

- For those aiming for a HD+ mark
- Everything taught in the course might be in these questions
- Will also test your ability to break a problem down into its parts
- The Prac Exam and past papers have an example of past Question 11 so you can see the difficulty level

Exam Marking

- Q1-4 and 9-11 are manually marked to award marks for correct code
- Q5-8 (debugging questions) are automarked only.
- Passing a certain number of autotests does not guarantee any specific mark
- There are extra tests used in automarking in addition to the ones you get given as autotests during the exam.

Exam Marking and Style

- There are no marks for style
  - So you don't need to explain your code in comments
  - But it needs to be readable so I can mark it
  - And so you can debug it and not get confused!!

The Exam: What should I study?

- The basics are important!
- Know how to use both arrays and linked lists
- Go back and do the **problem sets** if you haven't already or redo them for revision
- The **revision exercises** on the course webpage are also very useful to do and/or redo
- Try to do **coding exercises** from lectures without looking at the answers
- Try the **Week 8 lab exam** and **week 10 practice** exam!
- Try extra **practice exams** (coming out end of week 10)

The Exam: What should I study?

- Variables, Structs, enums, IF, Looping, Functions, Arrays, Linked Lists are very important to understand!
- You will need to have some understanding of Strings, Pointers, and Memory Allocation to be able to work successfully with char arrays, and linked lists
- Advice: Stop using chatgpt as a crutch if you are doing so
    - You will not have that in the exam

Log in Details

- Make sure you know you zid and zpass to log into the exam environment.
- You will not have access to phones or internet or paper notes to look up these log-in details!
- Please make sure you have them memorised

The Exam: Hints and Tips

- Read all questions before starting
- Start with the easier questions
- Make sure you read the question before jumping in and coding up something the question is not asking or using libraries it says not to use.
- Prepare! A couple of minutes thinking and drawing a diagram will clarify how you're going to approach a question
- When you are struggling with a question (particularly linked lists) DRAW DIAGRAMS!

The Exam: Hints and Tips

- Less questions answered completely is better than more questions partially answered
  - But don’t get stuck on a question you can’t make progress on for the whole 3 hours!!!
- The later questions have been designed to be very challenging
  - You do not need to complete them to get a great mark in this subject

The Exam: Advice

- Try to get a good night's sleep
- Try to eat properly before so you can concentrate.
- Make sure you breathe!
- You can do it!



What did we learn today?

- Finished Email Management System
- Exam Details!!!
- Revision Exercises

Next Week

- Revision
- Q&A with tutors at the end of the last lecture

Feedback Please!

Your feedback is valuable!

If you have any feedback from today's lecture, please follow the link below or use the QR Code.

Please remember to keep your feedback constructive, so I can action it and improve your learning experience.



https://forms.office.com/r/3M9hVZWL2Z

Reach Out

Content Related Questions:

Course Forum

Admin related Questions email:
cs1511@unsw.edu.au



Student Support - I Need Help With...

My Feelings and Mental Health Feeling depressed, overwhelmed or not your usual self?	**Mental Health Connect** student.unsw.edu.au/**mhc** **TalkCampus** student.unsw.edu.au/**Talkcampus**	**UNSW (24/7) Mental Health Support:** +61(2) 9385 5418 **Text support After Hours**: 0485 826 595 **Offshore** **Call 24-hour Medibank Hotline:** +61 2 8905 0307
Uni and Life in Australia Stress, financial, visas, Accommodation & More	**Student Support** **Indigenous Student Support**	student.unsw.edu.au/**advisors** nura-gili-centre-indigenous-programs
Reporting Sexual Assault/Harassment	**Equity Diversity & Inclusion (EDI)**	edi.unsw.edu.au/**sexual-misconduct**
Educational Adjustments To manage my studies and disability / health condition	**Equitable Learning Services (ELS)**	student.unsw.edu.au/**els**
Academic and Study Skills	**Academic Skills**	student.unsw.edu.au/**skills**
Special Consideration Because life impacts our studies and exams	**Special Consideration**	student.unsw.edu.au/**special-consideration**
Physical Health **Doctors visits, Vaccinations**	**Health Service**	student.unsw.edu.au/**hsu**