COMP9020

Foundations of Computer Science

Lecture 1: Course Introduction

Pre-course polls

Pre-course questionnaire

Pre-course poll

Acknowledgement of Country

I would like to acknowledge and pay my respect to the Bedegal people who are the Traditional Custodians of the land on which UNSW is built, and of Elders past and present.

Outline

Who are we?

Why are we here?

How will you be assessed?

What do I expect from you?

COMP9020 23T2 Staff

Lecturer:	Paul Hunter
Email:	paul.hunter@unsw.edu.au
Lectures:	Mondays 12-2pm and Wednesdays 12-2pm
Consults:	Thursdays and Sundays 7-8pm (online)
Research:	Theoretical CS: Algorithms, Formal verification
Consultations:	Mark Raya, Ziming Gong, Malhar Patel

Interactions

Lectures:

- Recordings available on echo360 (through Moodle)

Consultations:

- Microsoft Teams
- Group-based, student-driven
- Wiki for questions

Other points of contact:

- Formatif Learning Environment
- Course forums (ed)
- Email
- Weekly feedback

Outline

Who are we?

Why are we here?

How will you be assessed?

What do I expect from you?

What is this course about?

What is Computer Science?

What is this course about?

What is Computer Science?

> *"Computer science no more about computers than astronomy is about telescopes"*
>
> – E. Dijkstra

Course Aims

Computer Science is about

Course Aims

Computer Science is about exploring the ability, and limitation, of computers to solve problems. It covers:

- **What** are computers capable of solving?
- **How** can we get computers to solve problems?
- **Why** do these approaches work?

This course aims to increase your level of mathematical maturity to assist with the fundamental problem of **finding, formulating, and proving** properties of programs.

Key skills you will learn:

- Working with abstract concepts
- Giving logical (and rigorous) justifications
- Formulating problems so they can be solved computationally

Course Goals

By the end of the course, you should know enough to **understand** the answers to questions like:

What other questions would you like to know the answer to?

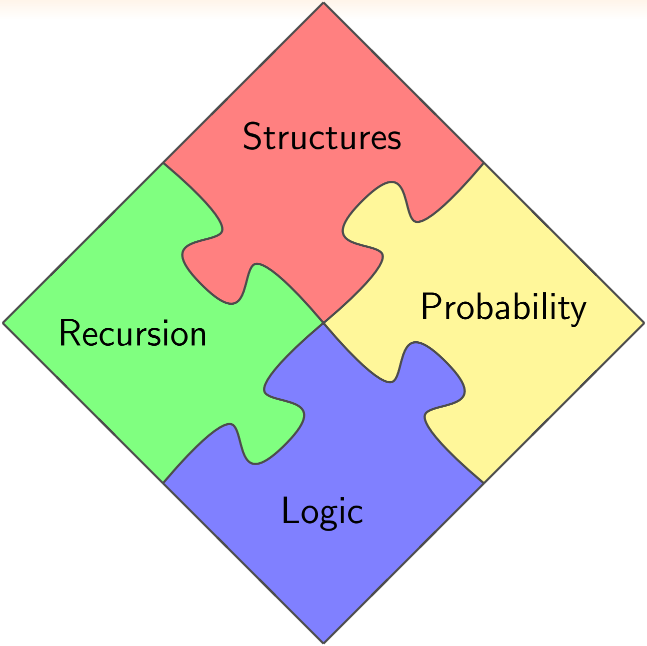
Course Goals

By the end of the course, you should know enough to **understand** the answers to questions like:

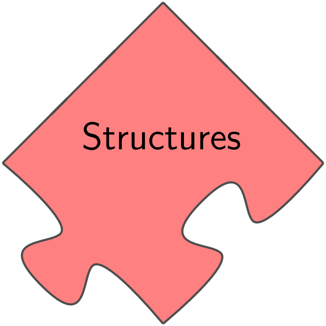
- How does RSA encryption work?
- Why do we use Relational Databases?
- How does Deep Learning work?
- Can computers think?
- How do Quantum Computers work?

What other questions would you like to know the answer to?

Course Topics

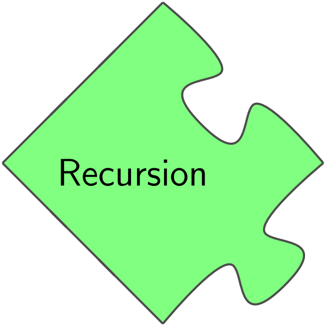


Course Topics



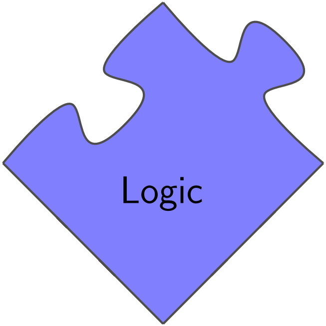
- Week 2: Set Theory
- Week 2: Formal Languages
- Week 3: Relations
- Week 4: Functions
- Week 5: Graph Theory

Course Topics



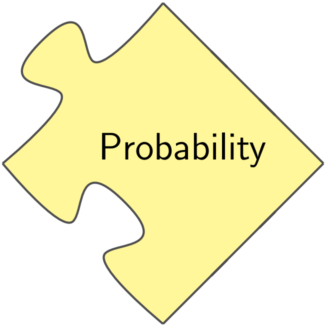
- Week 6: Recursion
- Week 7: Algorithmic Analysis
- Week 7: Induction

Course Topics



- Week 8: Boolean Logic
- Week 8: Propositional Logic

Course Topics



- Week 9: Combinatorics
- Week 9: Probability
- Week 10: Statistics

Course Material

All course information is placed on the course website

www.cse.unsw.edu.au/∼cs9020/

Content includes:

- Lecture slides and recordings
- Quizzes and Assignments
- Course Forums
- Practice questions
- Challenge questions

Course Material

Textbooks:

- KA Ross and CR Wright: Discrete Mathematics
- E Lehman, FT Leighton, A Meyer:
  Mathematics for Computer Science

Alternatives:

- K Rosen: Discrete Mathematics and its Applications

Outline

Assessment Philosophy

What is the purpose of assessment?

Two types of assessment:

- Formative: Formatif Tasks
- Summative: Quizzes, Assignments, Exam

Assessment Summary

60% exam, 30% assignments, 10% quizzes:

- Weekly Formatif tasks, worth up to 10 marks in total
- 10 quizzes, worth up to 1.25 marks each
  - Each Quiz: 4-6 threshold questions; 4-6 mastery questions
- 2 assignments, worth up to 10 marks each
- final exam (3 hours) worth up to 60 marks

Quizzes are available for 48 hours before the first lecture of the week. Assignments due on Fridays of weeks 6 and 10.

You must achieve 40% on the final exam to pass

Your final score will be taken from your 8 best quiz results, 2 assignments, Formatif portfolio and final exam.

Formative vs Summative assessment

Formative assessments:

- Students select level of difficulty (can be changed)
- Complete weekly tasks to a satisfactory level
- Regular feedback from teaching staff
- At the end of term, complete a portfolio to justify your grade

Summative assessments:

- Assess a broader selection of topics
- Quizzes test weekly concepts; Assignments cover multiple topics
- Marked against criteria

Late policy and Special Consideration

All assessments are submitted through the course website

Lateness policy

- Assignments: 5% of total grade off raw mark per 24 hours or part thereof
- Formatif tasks: No penalty but extensive extension requests will be noted
- Quizzes: Late submissions not accepted
- Exam: Late submissions not accepted

If you cannot meet a deadline through illness or misadventure you need to apply for Special Consideration.

More information

View the course outline at:

https://webcms3.cse.unsw.edu.au/COMP9020/23T3/outline

Particularly the sections on **Student conduct** and **Plagiarism**.

Outline

Who are we?

Why are we here?

How will you be assessed?

What do I expect from you?

Learning Objectives

I am always looking for you to **demonstrate**:

- Your understanding of the material
- Your ability to work with the material

NB

How you get an answer is as, if not more important than what the answer is.

Why?

Mathematical communication

Guidelines for good mathematical writing

Mathematical writing should be:

- Clear
- Logical
- Convincing

NB

All submitted work must be typeset. Diagrams may be hand drawn.

How can you do well?

The best way to improve is to **practice**.

Opportunities for you:

- Formatif tasks - proof-based questions in an environment for providing quick feedback
- Practice questions – including past exam questions
  - Looking for solutions! (Post to forum)
- Challenge questions
- Textbook and other questions (links on the course website)

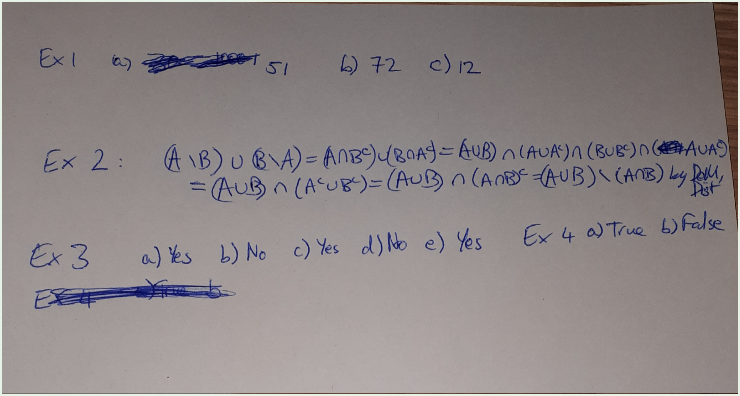
Opportunities from you:

- Post questions to the forum
- Bring questions to the lectures and/or consultations

I am always looking for more questions!

Examples

Example (Bad)



Examples

Example (Good)

Ex. 2

$$
\begin{aligned}
(A \setminus B) \cup (B \setminus A) &= (A \cap B^c) \cup (B \cap A^c) && \text{(Def.)} \\
&= ((A \cap B^c) \cup B) \cap ((A \cap B^c) \cup A^c) && \text{(Dist.)} \\
&= (A \cup B) \cap (B^c \cup B) \\
&\quad \cap (A \cup A^c) \cap (B^c \cup A^c) && \text{(Dist.)} \\
&= (A \cup B) \cap (A^c \cup B^c) && \text{(Ident.)} \\
&= (A \cup B) \cap (A \cap B)^c && \text{(DeM.)} \\
&= (A \cup B) \setminus (A \cap B) && \text{(Def.)}
\end{aligned}
$$

Examples

Example (Good)

Ex. 4a

We will show that if R_1 and R_2 are symmetric, then $R_1 \cap R_2$ is symmetric.

Suppose $(a, b) \in R_1 \cap R_2$.

Then $(a, b) \in R_1$ and $(a, b) \in R_2$.

Because R_1 is symmetric, $(b, a) \in R_1$; and because R_2 is symmetric, $(b, a) \in R_2$.

Therefore $(b, a) \in R_1 \cap R_2$.

Therefore $R_1 \cap R_2$ is symmetric.

Proofs

A large component of your work in this course is giving **proofs** of **propositions.**

A **proposition** is a statement that is either true or false.

Example

Propositions:

- $3 + 5 = 8$
- All integers are either even or odd
- There exist a, b, c such that $1/a + 1/b + 1/c = 4$

Not propositions:

- $3 + 5$
- x is even or x is odd
- $1/a + 1/b + 1/c = 4$

Proposition structure

Common proposition structures include:

If A then B	$(A \Rightarrow B)$
A if and only if B	$(A \Leftrightarrow B)$
For all x, A	$(\forall x.A)$
There exists x such that A	$(\exists x.A)$

$\forall$ and $\exists$ are known as **quantifiers**.

Proofs

A large component of your work in this course is giving **proofs** of **propositions.**

A proof of a proposition is an argument to convince the reader/marker that the proposition is true.

A **proof** of a proposition is a finite sequence of logical steps, starting from base assumptions (**axioms** and **hypotheses**), leading to the proposition in question.

Proofs

Example

Prove: $3 \times 2 = 2 \times 3$

$$\begin{aligned}
3 \times 2 &= (2+1) \times 2 \\
&= (2 \times 2) + (1 \times 2) \\
&= (1 \times 2) + (2 \times 2) \\
&= 2 + (2 \times 2) \\
&= (2 \times 1) + (2 \times 2) \\
&= 2 \times (1 + 2) \\
&= 2 \times 3.
\end{aligned}$$

Proofs: How much detail?

- Depends on the context (question, expectation, audience, etc)
- Each **step** should be justified (excluding basic algebra and arithmetic)

Guiding principle

Proofs should demonstrate your **ability** and your **understanding**.

Proofs: pitfalls

Starting from the proposition and deriving true **is not valid**.

Example

Prove: $0 = 1$

$$\begin{array}{rrcl} & 0 & = & 1 \\ \text{So (mult. by 2)} & 0 & = & 2 \\ \text{So (subtract 1)} & -1 & = & 1 \\ \text{So} & (-1)^2 & = & (1)^2 \\ \text{So} & 1 & = & 1 \quad \text{which is true.} \end{array}$$

Does this mean that $0 = 1$?

Proofs: pitfalls

Make sure each step is logically valid

Example

$$
\begin{array}{lrcl}
 & -20 & = & -20 \\
\text{So} & 25-45 & = & 16-36 \\
\text{So} & 5^2-2\cdot 5\cdot \frac{9}{2} & = & 4^2-2\cdot 4\cdot \frac{9}{2} \\
\text{So} & 5^2-2\cdot 5\cdot \frac{9}{2}+\left(\frac{9}{2}\right)^2 & = & 4^2-2\cdot 4\cdot \frac{9}{2}+\left(\frac{9}{2}\right)^2 \\
\text{So} & (5-\frac{9}{2})^2 & = & (4-\frac{9}{2})^2 \\
\text{So} & 5-\frac{9}{2} & = & 4-\frac{9}{2}
\end{array}
$$

Does this mean that $5=4$?

Proofs: pitfalls

Make sure each step is logically valid

Example

Suppose $a = b$. Then,

$$\begin{array}{lrcl} & a^2 & = & ab \\ \text{So} & a^2 - b^2 & = & ab - b^2 \\ \text{So} & (a-b)(a+b) & = & (a-b)b \\ \text{So} & a+b & = & b \\ \text{So} & a & = & 0 \end{array}$$

This is true no matter what value a is given at the start, so does that mean everything is equal to 0?

Proofs: pitfalls

For propositions of the form $\forall x.A$ where x can have infinitely many values:

- You cannot enumerate infinitely many cases in a proof.
- Only considering a finite number of cases is not sufficient.

Example

$$\text{For all } n,\ n^2 + n + 41 \text{ is prime}$$

True for $n = 0, 1, 2, \ldots, 39$. Not true for $n = 40$.

Proofs: pitfalls

The order of quantifiers matters when it comes to propositions:

Example

- For every number x, there is a number y such that y is larger than x
- There is a number y such that for every number x, y is larger than x

Proof strategies: direct proof

Proposition form	You need to do this
$A \Rightarrow B$	Assume A and prove B
$A \Leftrightarrow B$	Prove "If A then B" and "If B then A"
$\forall x.A$	Show A holds for every possible value of x
$\exists x.A$	Find a value of x that makes A true

Proof strategies: contradiction

To prove A is true, assume A is false and derive a contradiction.
That is, start from the negation of the proposition and derive false.

Example

Prove: $\sqrt{2}$ is irrational
Proof: Assume $\sqrt{2}$ is rational ...

Negating propositions

Proposition form	Its negation
A and B	not A or not B
A or B	not A and not B
$A \Rightarrow B$	A and not B
$A \Leftrightarrow B$	A and not B, or B and not A
$\forall x.A$	$\exists x.$ not A
$\exists x.A$	$\forall x.$ not A

Proof strategies: contrapositive

To prove a proposition of the form "If A then B" you can prove "If not B then not A"

Example

Prove: If $m + n \geq 73$ then $m \geq 37$ or $n \geq 37$.

Proof strategies: dealing with $\forall$

How can we check infinitely many cases?

- Choose an **arbitrary** element: an object with no assumptions about it (may have to check several cases)
- Induction (see week 5)

Example

Prove: For every integer n, n^2 will have remainder 0 or 1 when divided by 4.

Note: "Arbitrary" is not the same as "random".