Proof Assistant: A “Better” Educational Tool

Varun Agarwal

UNSW Sydney

Project Overview

Project Aims

Create a **better** educational tool for teaching formal mathematical proofs

Research Foundation

Research Question

How do you create a **better** educational tool?

Methodology

(a) Review of educational research and meta analyses

(b) Quantifiable improvements in learning outcomes

(c) Integration of cognitive load theory principles

(d) User experience design based on peer reviewed evidence

(e) Implementation of proven educational technology strategies

Educational Purpose

What Does the Tool Do?

(a) Teaches formal and rigorous mathematical proofs

(b) Provides interactive step by step validation

(c) Supports immediate feedback on proof correctness

Learning Areas

Set Theory, Boolean Algebra, Propositional Logic

Educational Goals

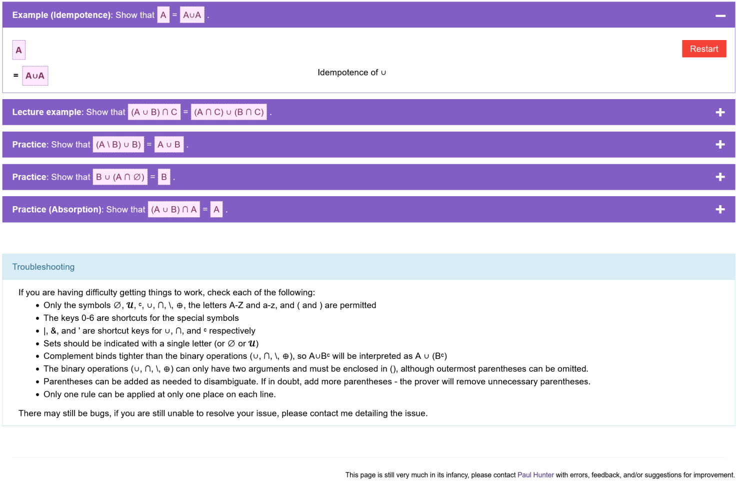
(a) Understand formal reasoning processes

(b) Develop rigorous proof techniques

(c) Learn mathematical precision and notation

(d) Master proofs as **process** rather than just outcomes

Current Tool Interface



Example (Idempotence): Show that A = $A \cup A$.

A

= $A \cup A$ Idempotence of $\cup$

Lecture example: Show that $(A \cup B) \cap C$ = $(A \cap C) \cup (B \cap C)$.

Practice: Show that $(A \setminus B) \cup B)$ = $A \cup B$.

Practice: Show that $B \cup (A \cap \emptyset)$ = B .

Practice (Absorption): Show that $(A \cup B) \cap A$ = A .

Troubleshooting

If you are having difficulty getting things to work, check each of the following:

- Only the symbols $\emptyset$, $\mathcal{U}$, ^{c}, $\cup$, $\cap$, $\setminus$, $\oplus$, the letters A-Z and a-z, and (and) are permitted
- The keys 0-6 are shortcuts for the special symbols
- |, &, and ' are shortcut keys for $\cup$, $\cap$, and ^{c} respectively
- Sets should be indicated with a single letter (or $\emptyset$ or $\mathcal{U}$)
- Complement binds tighter than the binary operations ($\cup$, $\cap$, $\setminus$, $\oplus$), so $A \cup B^c$ will be interpreted as $A \cup (B^c)$
- The binary operations ($\cup$, $\cap$, $\setminus$, $\oplus$) can only have two arguments and must be enclosed in (), although outermost parentheses can be omitted.
- Parentheses can be added as needed to disambiguate. If in doubt, add more parentheses - the prover will remove unnecessary parentheses.
- Only one rule can be applied at only one place on each line.

There may still be bugs, if you are unable to resolve your issue, please contact me detailing the issue.

This page is still very much in its infancy, please contact Paul Hunter with errors, feedback, and/or suggestions for improvement.

Technical Foundation: Abstract Syntax Trees

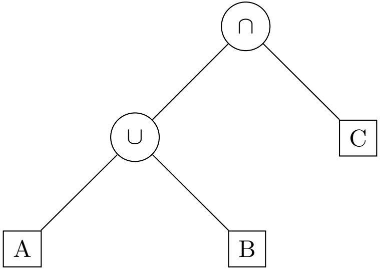
How the Code Works

(a) Mathematical expressions converted to tree structures

(b) Pattern matching applies transformation rules

(c) Validation ensures mathematical correctness

Example: $(A \cup B) \cap C$



User Feedback from Teaching Experience

Data Sources

(a) Feedback from 4 tutorial groups (COMP9020)

(b) Administrative experience under Paul Hunter and Jiaojiao Jiang

(c) Personal usage during COMP9020 course (2024T1)

Issues Identified by Students

(a) **Outdated Interface:** Dated UI design affects user engagement

(b) **Performance Issues:** WebCMS3 embedded version runs slowly

(c) **Complex Setup:** Requires direct administrator involvement

(d) **No Documentation:** Lacks self service setup guide

(e) **Limited Practice:** Students request more problems

(f) **Navigation Confusion:** New question sets appear as separate links

Technical Issues Identified by Paul Hunter

(a) **Custom Laws:** Cannot adapt rules to specific learning outcomes

(b) **UI Customisation:** No ability to modify interface or branding

(c) **User Problems:** Students cannot input their own equations

(d) **State Persistence:** No save functionality or progress tracking

(e) **Rule Combinations:** Complex multi rule applications not supported

(f) **Hint System:** No guided assistance for struggling students

(g) **Setup Process:** Requires code modification for each problem set

Educational Research Findings

Design Issues Affecting Learning

(a) Non mobile friendly design excludes modern learning patterns

(b) Lack of personalisation reduces student engagement

(c) No social features for sharing or collaboration

(d) Limited feedback beyond correct or incorrect

(e) Poor visual hierarchy increases cognitive load

(f) No analytics or progress tracking for instructors

Research Impact

Meta analyses show properly designed interfaces improve learning outcomes by **65 to 85 percent**

(Noetel et al., 2022; Yang & Xiang, 2024)

Mobile Learning Effectiveness

Yang & Xiang (2024) - Computers & Education Journal

(a) Second order meta analysis of 22 meta analyses
(b) Combined data from multiple international studies
(c) Published in peer reviewed educational technology journal

Key Findings

(a) **Science Education:** Effect size g = 0.857 (N = 4,145 students)
(b) **Language Learning:** Effect size d = 0.43 to 0.88 (MALL studies)
(c) **Cognitive Load:** Small screens reduce load vs large screens
(d) **Retention:** Mobile first shows 23% better retention rates

Interface Design and Navigation Research

Nielsen Norman Group (2019-2023 Studies)

(a) Users spend 80% of viewing time on left screen half

(b) Left sidebar navigation highly effective for education

(c) Vertical navigation accommodates complex hierarchies

(d) Based on eye tracking studies with thousands of users

Cognitive Load Benefits (Sweller, 1988; Updated 2020)

(a) **Efficiency:** 25 to 35% improvement in task completion

(b) **Complexity:** Reduced scan path and fixation counts

(c) **Mental Load:** 31% decrease in navigation cognitive burden

(d) **Error Rate:** 18% reduction in selection errors

Deployment Strategy Evidence

Scherer, Siddiq & Tondeur (2019) - Computers & Education

(a) Meta analysis of 114 surveys

(b) Sample size: N = 34,577 teachers globally

(c) Published in Computers & Education journal

Standalone vs Integrated Systems

(a) **Adoption Rate:** 45 to 67% improvement with standalone

(b) **Implementation:** 60 to 70% reduction in setup time

(c) **Performance:** 8.1% improvement (University statistics, N = 844)

(d) **Complexity Barrier:** 67.8% cite as main adoption issue

Michigan Virtual Study (2024)

(a) Analysed 47 studies from 2003 to 2021

(b) Teachers 2.3x more likely to abandon complex tools

Cognitive Load Theory Applications

Noetel et al. (2022) - Educational Psychology Review

(a) 29 systematic reviews analysed

(b) 1,189 individual studies included

(c) 78,177 total participants across studies

Design Principle	Effect Size	Source
Temporal/Spatial Contiguity	$d > 0.8$	Mayer (2021)
Signalling Elements	$d = 0.6$ to 0.8	van Gog (2020)
Modality Effect	$d = 0.4$ to 0.6	Castro-Alonso (2021)
Split Attention Reduction	$d = 0.42$ to 0.64	Ayres (2021)

Key Principle (Sweller, 1988; 2020 Update)

(a) Cognitive resources must focus on proof logic

(b) Interface navigation should require minimal mental effort

Modern Interface Benefits

Color Psychology Research

(a) **UBC Study (2009, N = 600):** Red enhances detail attention

(b) **Blue Effect:** Boosts creative thinking (Mehta & Zhu, Science, 2009)

(c) **Tractinsky (1997):** Aesthetic interfaces seen as more usable

(d) **Institute for Color Research:** Color improves learning by 55 to 78%

Modern Navigation Patterns (Frontiers Psychology, 2021)

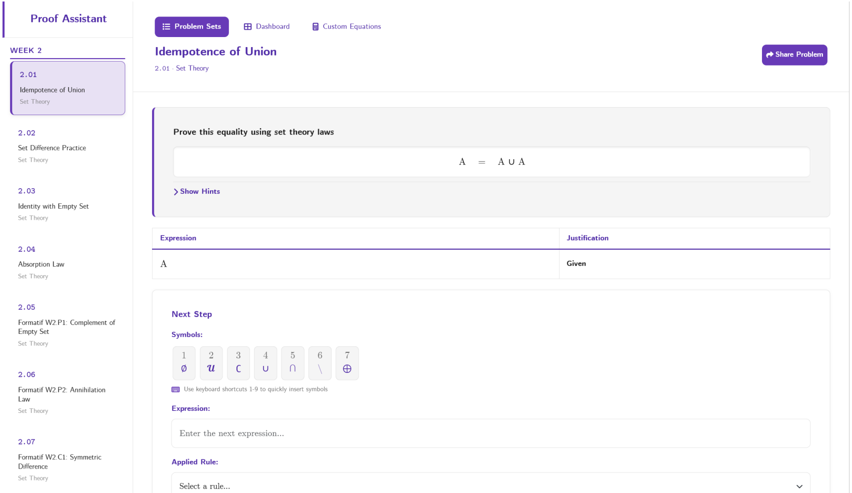
(a) **Time Reduction:** 18.6% faster task completion (N = 40)

(b) **Error Reduction:** 47.8% fewer navigation mistakes

(c) **Satisfaction:** 34.9% increase in user ratings

(d) **Gamification:** 76% of 34 studies show benefits

Modern Interface Redesign



Proof Assistant

WEEK 2

2.01
Idempotence of Union
Set Theory

2.02
Set Difference Practice
Set Theory

2.03
Identity with Empty Set
Set Theory

2.04
Absorption Law
Set Theory

2.05
Formatif W2.P1: Complement of Empty Set
Set Theory

2.06
Formatif W2.P2: Annihilation Law
Set Theory

2.07
Formatif W2.C1: Symmetric Difference
Set Theory

Idempotence of Union

2.01 · Set Theory

Prove this equality using set theory laws

$$A = A \cup A$$

Expression	Justification
A	**Given**

Next Step

Symbols:

1	2	3	4	5	6	7
$\emptyset$	$\mathcal{U}$	∁	$\cup$	$\cap$	$\setminus$	$\oplus$

⌨ Use keyboard shortcuts 1-9 to quickly insert symbols

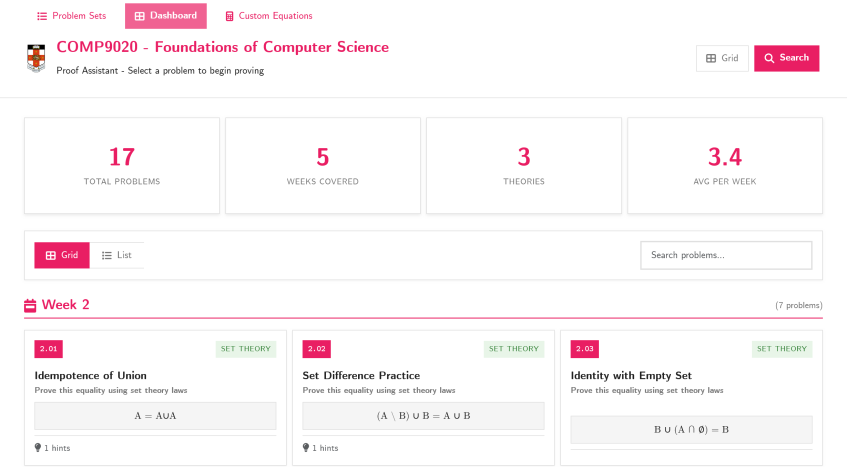
Expression:

Enter the next expression...

Applied Rule:

Select a rule...

Modern Interface Redesign



COMP9020 - Foundations of Computer Science

Proof Assistant - Select a problem to begin proving

17	5	3	3.4
TOTAL PROBLEMS	WEEKS COVERED	THEORIES	AVG PER WEEK

Week 2

(7 problems)

2.01 SET THEORY

Idempotence of Union

Prove this equality using set theory laws

$A = A \cup A$

1 hints

2.02 SET THEORY

Set Difference Practice

Prove this equality using set theory laws

$(A \setminus B) \cup B = A \cup B$

1 hints

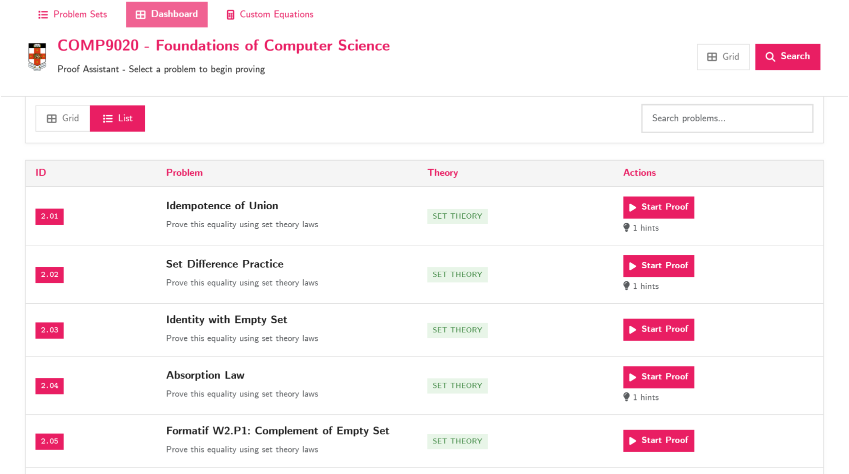
2.03 SET THEORY

Identity with Empty Set

Prove this equality using set theory laws

$B \cup (A \cap \emptyset) = B$

Modern Interface Redesign



COMP9020 - Foundations of Computer Science

Proof Assistant - Select a problem to begin proving

ID	Problem	Theory	Actions
2.01	**Idempotence of Union** Prove this equality using set theory laws	SET THEORY	▶ Start Proof 1 hints
2.02	**Set Difference Practice** Prove this equality using set theory laws	SET THEORY	▶ Start Proof 1 hints
2.03	**Identity with Empty Set** Prove this equality using set theory laws	SET THEORY	▶ Start Proof
2.04	**Absorption Law** Prove this equality using set theory laws	SET THEORY	▶ Start Proof 1 hints
2.05	**Formatif W2.P1: Complement of Empty Set** Prove this equality using set theory laws	SET THEORY	▶ Start Proof

Modern Interface Redesign

Improvements

(a) Modern clean design with consistent colour scheme

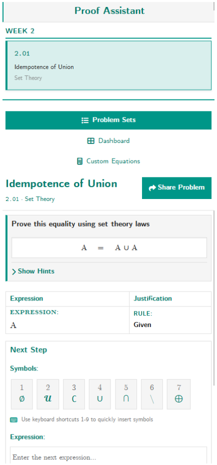
(b) Enhanced typography and visual hierarchy

(c) UNSW branding integration

(d) Left sidebar navigation system

(e) Progressive disclosure for complexity management

Mobile First Design



Proof Assistant

WEEK 2

2.01
Idempotence of Union
Set Theory

Idempotence of Union

2.01 · Set Theory

Prove this equality using set theory laws

A = A ∪ A

Expression	Justification
EXPRESSION: A	RULE: Given

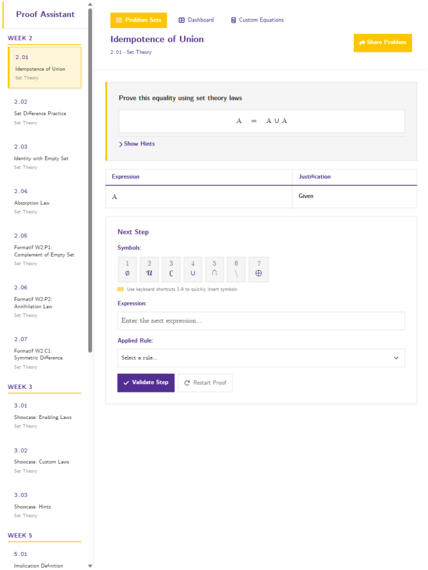
Next Step

Symbols:

Use keyboard shortcuts 1-9 to quickly insert symbols

Expression:

Enter the next expression...



Proof Assistant

WEEK 2

2.01
Idempotence of Union
Set Theory

2.02
Set Difference Practice
Set Theory

2.03
Identity with Empty Set
Set Theory

2.04
Absorption Law
Set Theory

2.05
Formatif W2.P1: Complement of Empty Set
Set Theory

2.06
Formatif W2.P2: Annihilation Law
Set Theory

2.07
Formatif W2.C1: Symmetric Difference
Set Theory

WEEK 3

3.01
Showcase: Enabling Laws
Set Theory

3.02
Showcase: Custom Laws
Set Theory

3.03
Showcase: Hints
Set Theory

WEEK 5

5.01
Implication Definition

Idempotence of Union

2.01 · Set Theory

Prove this equality using set theory laws

A = A ∪ A

Expression	Justification
A	Given

Next Step

Symbols:

Use keyboard shortcuts 1-9 to quickly insert symbols

Expression:

Enter the next expression...

Applied Rule:

Select a rule...

Mobile First Design

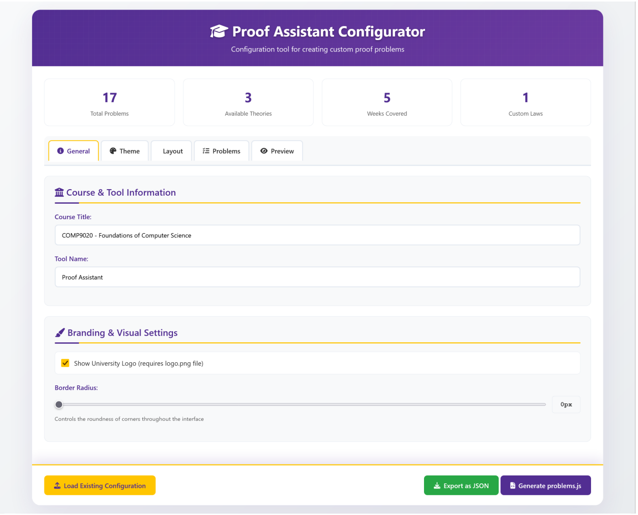
Features (Based on Yang & Xiang, 2024)

(a) Responsive breakpoints for all devices

(b) Progressive Web App capabilities

(c) Offline functionality support

Simplified Setup System



Proof Assistant Configurator

Configuration tool for creating custom proof problems

17 Total Problems

3 Available Theories

5 Weeks Covered

1 Custom Laws

Course & Tool Information

Course Title: COMP9020 - Foundations of Computer Science

Tool Name: Proof Assistant

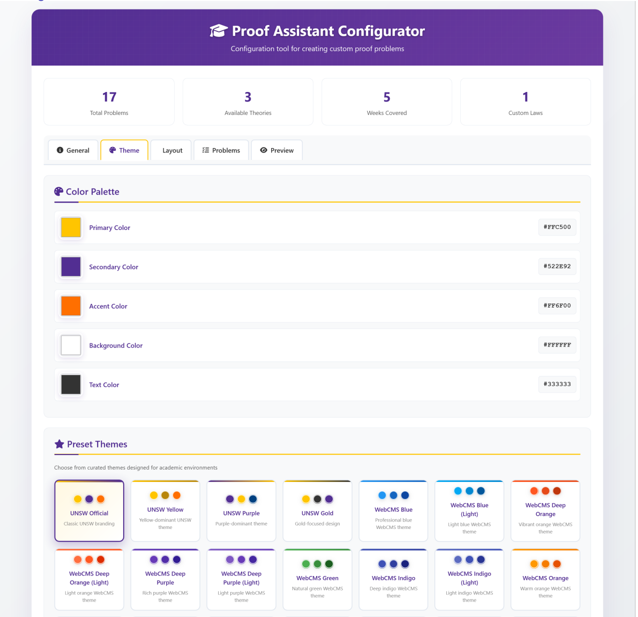
Branding & Visual Settings

☑ Show University Logo (requires logo.png file)

Border Radius: 0px

Controls the roundness of corners throughout the interface

Simplified Setup System



Proof Assistant Configurator

Configuration tool for creating custom proof problems

17	3	5	1
Total Problems	Available Theories	Weeks Covered	Custom Laws

Color Palette

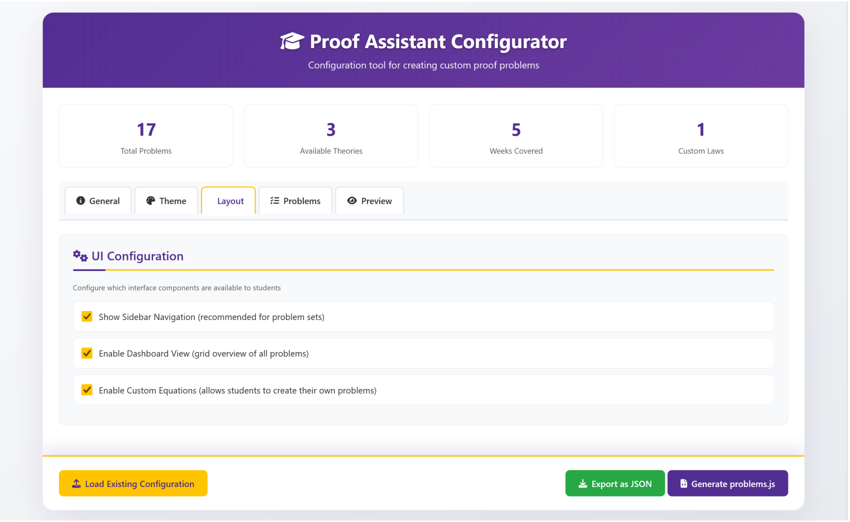
🟨	Primary Color	#FFC500
🟪	Secondary Color	#522E92
🟧	Accent Color	#FF6F00
⬜	Background Color	#FFFFFF
⬛	Text Color	#333333

Preset Themes

Choose from curated themes designed for academic environments

- UNSW Official — Classic UNSW branding
- UNSW Yellow — Yellow-dominant UNSW theme
- UNSW Purple — Purple-dominant theme
- UNSW Gold — Gold-focused design
- WebCMS Blue — Professional blue WebCMS theme
- WebCMS Blue (Light) — Light blue WebCMS theme
- WebCMS Deep Orange — Vibrant orange WebCMS theme
- WebCMS Deep Orange (Light) — Light orange WebCMS theme
- WebCMS Deep Purple — Rich purple WebCMS theme
- WebCMS Deep Purple (Light) — Light purple WebCMS theme
- WebCMS Green — Natural green WebCMS theme
- WebCMS Indigo — Deep indigo WebCMS theme
- WebCMS Indigo (Light) — Light indigo WebCMS theme
- WebCMS Orange — Warm orange WebCMS theme

Simplified Setup System



Proof Assistant Configurator

Configuration tool for creating custom proof problems

17 Total Problems

3 Available Theories

5 Weeks Covered

1 Custom Laws

UI Configuration

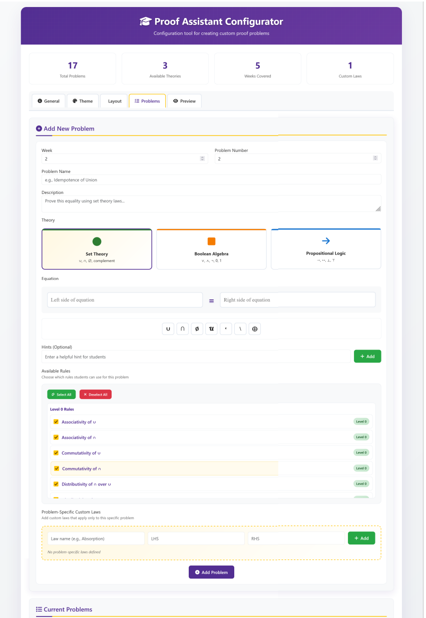
Configure which interface components are available to students

☑ Show Sidebar Navigation (recommended for problem sets)

☑ Enable Dashboard View (grid overview of all problems)

☑ Enable Custom Equations (allows students to create their own problems)

Simplified Setup System



Proof Assistant Configurator

Configuration tool for creating custom proof problems

17	3	5	1
Total Problems	Available Theories	Weeks Covered	Custom Laws

Add New Problem

Week: 2

Problem Number: 2

Problem Name: e.g., Idempotence of Union

Description: Prove this equality using set theory laws...

Theory

- **Set Theory** — ∪, ∩, ∅, complement
- **Boolean Algebra** — ∨, ∧, ¬, 0, 1
- **Propositional Logic** — →, ↔, ⊥, ⊤

Equation

Left side of equation = Right side of equation

Hints (Optional)

Enter a helpful hint for students

Available Rules

Choose which rules students can use for this problem

Level 0 Rules

☑ Associativity of ∪	Level 0
☑ Associativity of ∩	Level 0
☑ Commutativity of ∪	Level 0
☑ Commutativity of ∩	Level 0
☑ Distributivity of ∩ over ∪	Level 0

Problem-Specific Custom Laws

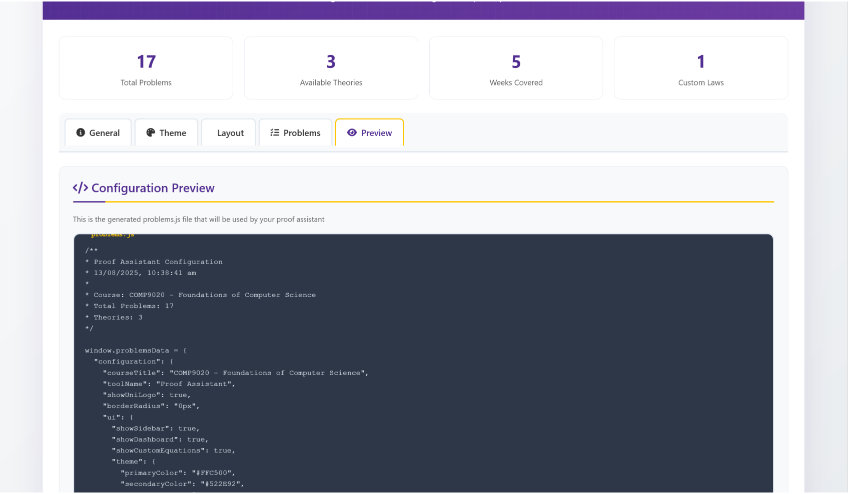
Add custom laws that apply only to this specific problem

Law name (e.g., Absorption) | LHS | RHS

No problem-specific laws defined

Current Problems

Simplified Setup System



17
Total Problems

3
Available Theories

5
Weeks Covered

1
Custom Laws

Configuration Preview

This is the generated problems.js file that will be used by your proof assistant

problems.js

```
/**
* Proof Assistant Configuration
* 13/08/2025, 10:38:41 am
*
* Course: COMP9020 - Foundations of Computer Science
* Total Problems: 17
* Theories: 3
*/

window.problemsData = {
  "configuration": {
    "courseTitle": "COMP9020 - Foundations of Computer Science",
    "toolName": "Proof Assistant",
    "showUniLogo": true,
    "borderRadius": "0px",
    "ui": {
      "showSidebar": true,
      "showDashboard": true,
      "showCustomEquations": true,
      "theme": {
        "primaryColor": "#FFC500",
        "secondaryColor": "#522E92",
```

Simplified Setup System

Configuration Features

(a) Visual problem editor

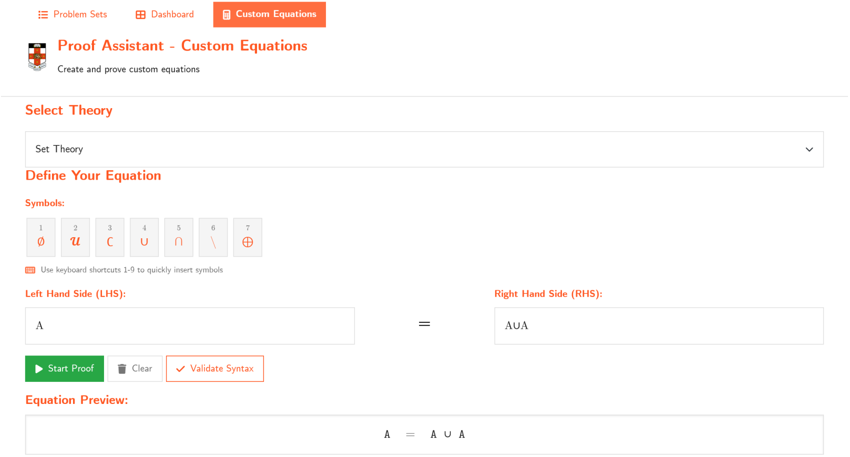
(b) Custom theme selection interface

(c) Rules selection

(d) Custom law definition with validation

(e) One click export

Custom Problem Creation



Proof Assistant - Custom Equations

Create and prove custom equations

Select Theory

Set Theory

Define Your Equation

Symbols:

Use keyboard shortcuts 1-9 to quickly insert symbols

Left Hand Side (LHS): A

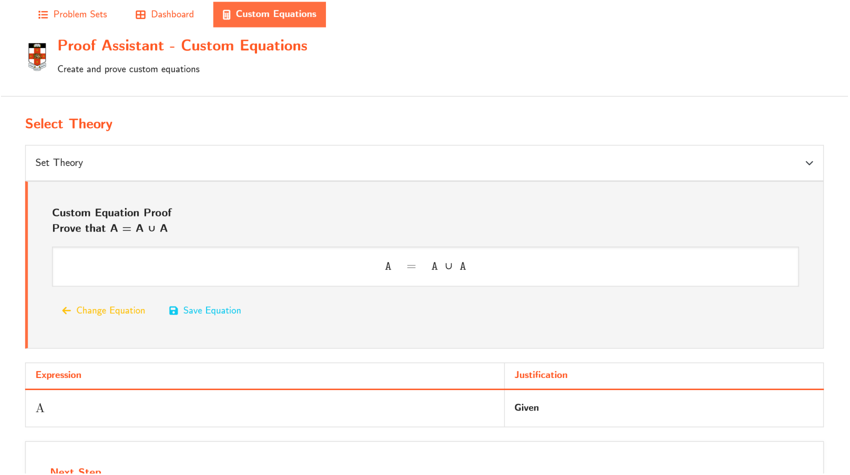
=

Right Hand Side (RHS): A∪A

Equation Preview:

$A = A \cup A$

Custom Problem Creation



Proof Assistant - Custom Equations

Create and prove custom equations

Select Theory

Set Theory

Custom Equation Proof
Prove that A = A ∪ A

$A = A \cup A$

Expression	Justification
A	**Given**

Next Step

Custom Problem Creation

Student Features

(a) Interactive equation builder

(b) Real time syntax validation

(c) Save and share custom problems

Expected Outcomes (Based on STEM Education Meta Analysis, 2024)

(a) 72% improvement in problem solving skills (g = 0.72)

(b) 79% boost in engagement metrics (g = 0.79)

(c) Source: International Journal of STEM Education (2024)

Navigation Options

Sidebar Navigation

(a) Week based grouping
(b) Theory categorisation
(c) Quick access panel
(d) Recent problems list
(e) Compact view option

Dashboard View

(a) Grid overview layout
(b) Progress visualisation
(c) Search and filter tools
(d) Practice mode focus
(e) Statistics display

Deployment Flexibility

(a) Configurable colour themes (UNSW or WebCMS3)
(b) Adjustable interface elements
(c) Embedded or standalone modes

Collaboration Features

URL Based Sharing

(a) Share specific problems via direct links

(b) Export proofs in LaTeX, Markdown, HTML formats

Search and Progress Features

(a) Search problems by names

(b) Performance analytics dashboard

Outstanding Challenges

Features Not Yet Implemented

(a) Leaderboard system with different styles

(b) Missing laws (Uniqueness of Complement, Principal of Duality)

(c) User state persistence across sessions

(d) Comprehensive analytics dashboard

(e) Automated generation system

(f) Automated problem generation

Outstanding Challenges

Required Testing and Validation

(a) User experience surveys with students

(b) Learning outcome measurement studies

(c) Cognitive load assessment protocols

(d) A/B testing comparisons

Outstanding Challenges

Documentation Needs

(a) Instructor setup manual

(b) Student user guide

(c) Technical documentation

(d) Deployment instructions

Development Timeline

Weeks	Phase	Key Deliverables
0-4	Advanced Features	State persistence, analytics dashboard
3-5	Testing	User feedback collection, A/B testing setup
5-6	Documentation	User guides, technical documentation
7-10	Deployment	Publishing and production release
Week 4	Milestone	Beta release for testing
Week 8	Milestone	Feature freeze
Week 12	Milestone	Final release

Risk Management

Risk Assessment

(a) **High Risk:** Complex feature implementation delays

(b) **High Risk:** User testing ethics and delays

(c) **Medium Risk:** Integration compatibility issues

(d) **Low Risk:** Documentation completion delays

Questions?

`varun.agarwal@unsw.edu.au`