# COMP2620/6262 (Logic) Tutorial

Week 7

Semester 1, 2025

## Tutorial Quiz

In each tutorial, apart from week 2, there is a short quiz on skills practised in the previous tutorial. Your top 7 quiz attempts, out of the 9 available, will collectively count for 50% of your final mark.

This week's quiz is on **Logic4Fun**. You should come prepared to use your device to log in to Logic4Fun, and have already joined our class, following instructions in Wattle. During the test you may look up Logic4Fun documentation on the Logic4Fun website: in particular you might like to have https://logic4fun.cecs.anu.edu.au/guide/built-ins open in a tab.

Your tutor will hand out paper face down with the description of the puzzle you need to translate into Logic4Fun syntax, along with some instructions and hints. On your tutor's signal you can turn your paper over and you have **fifteen minutes** to perform this task in the Solver window and submit. You may submit as often as you like, but later submissions overwrite earlier submissions, so do not submit at any time after your tutor has asked you to finish the task. This applies even if the test is still technically open on the website. For full marks your solution must pass the 'Check Syntax' button, and if your modelling is correct, the 'Solve' button will produce exactly one solution. Partial marks may still be available if these tests do not pass. You may ignore any warning that does not prevent the 'Check Syntax' button from printing 'Syntax checked...OK'.

Make sure that you know the number of the tutorial you are attending, and submit to that tutorial's number, even if it is different to your usual tutorial number.

## This Week's Exercises

This tutorial involves the tableaux rules:

$$\frac{\mathbf{T} : \bot}{\times} \qquad \frac{\mathbf{T} : \neg\varphi}{\mathbf{F} : \varphi} \qquad \frac{\mathbf{F} : \neg\varphi}{\mathbf{T} : \varphi} \qquad \frac{\mathbf{T} : \varphi \lor \psi}{\mathbf{T} : \varphi \qquad \mathbf{T} : \psi} \qquad \frac{\mathbf{F} : \varphi \lor \psi}{\begin{array}{c}\mathbf{F} : \varphi \\ \mathbf{F} : \psi\end{array}}$$

$$\frac{\mathbf{T} : \varphi \land \psi}{\begin{array}{c}\mathbf{T} : \varphi \\ \mathbf{T} : \psi\end{array}} \qquad \frac{\mathbf{F} : \varphi \land \psi}{\mathbf{F} : \varphi \qquad \mathbf{F} : \psi} \qquad \frac{\mathbf{T} : \varphi \to \psi}{\mathbf{F} : \varphi \qquad \mathbf{T} : \psi} \qquad \frac{\mathbf{F} : \varphi \to \psi}{\begin{array}{c}\mathbf{T} : \varphi \\ \mathbf{F} : \psi\end{array}}$$

$$\frac{\mathbf{T} : \forall x\varphi}{\begin{array}{c}\mathbf{T} : \varphi[a_1/x] \\ \mathbf{T} : \varphi[a_2/x] \\ \vdots \\ \mathbf{T} : \varphi[a_n/x]\end{array}} \qquad \frac{\mathbf{F} : \exists x\varphi}{\begin{array}{c}\mathbf{F} : \varphi[a_1/x] \\ \mathbf{F} : \varphi[a_2/x] \\ \vdots \\ \mathbf{F} : \varphi[a_n/x]\end{array}}$$

where $a_1, \dots, a_n$ are all terms (= variables) in the tableau appearing free before or after this line. If no variables appear free before this line, the conclusion is $\varphi[a/x]$ only.

$$\frac{\mathbf{F} : \forall x\varphi}{\mathbf{F} : \varphi[a/x]} \qquad \frac{\mathbf{T} : \exists x\varphi}{\mathbf{T} : \varphi[a/x]}$$

where $a$ does not appear free earlier in the tableau.

Branching rules for helping to find finite satisfying models in the presence of quantifiers:

$$\frac{\mathbf{F} : \forall x\varphi}{\mathbf{F} : \varphi[a_1/x] \quad \cdots \quad \mathbf{F} : \varphi[a_n/x] \quad \mathbf{F} : \varphi[a/x]} \qquad \frac{\mathbf{T} : \exists x\varphi}{\mathbf{T} : \varphi[a_1/x] \quad \cdots \quad \mathbf{T} : \varphi[a_n/x] \quad \mathbf{T} : \varphi[a/x]}$$

where $a_1, \ldots, a_n$ are all terms (= variables) in the tableau appearing free before or after this line, and $a$ does not appear free earlier in the tableau.

1. **The test at the start of the next tutorial will resemble this question. It will ask you to build a complete tableau for a collection of signed propositions (not first order logic). You will not be asked to explicitly extract a satisfying model.**

   Build complete tableaux for each of the following sequents and/or collections of signed propositions. Each branch of each tableaux should be completed, either closed (a contradiction) or open (no contradiction, but no more applicable rules)

   - $\neg(p \to q) \vdash p \to \neg q$
   - $p \to \neg q \vdash \neg(p \to q)$
   - $p \to \neg q, \neg q \to p \vdash \neg(p \to q) \vee \neg(q \to p)$
   - $\mathbf{T} : \bot \vee p, \mathbf{F} : p \wedge \neg q, \mathbf{F} : q \wedge r$

2. For each of the last group of questions: if it was a sequent, is it valid? Is the set of signed propositions (either given directly, or derived from a sequent) satisfiable? If it is satisfiable, what values for the propositions satisfies it?

3. Exclusive or, sometimes written with the symbol $\oplus$, is the propositional connective defined by $\varphi \oplus \psi$ being true if exactly one of $\varphi$ and $\psi$ are true. Propose tableaux rules for $\oplus$. Verify them by proving

   - $p \oplus q \vdash (p \vee q) \wedge \neg(p \wedge q)$
   - $(p \vee q) \wedge \neg(p \wedge q) \vdash p \oplus q$

4. Build complete tableaux for the following first order logic sequents or sets of signed formulas. Do not use the branching quantifier rules. If the signed formulas are satisfiable, give a satisfying model.

   - $\forall x(F\,x \to H\,x), \exists x(G\,x \wedge \neg H\,x) \vdash \neg\forall x(G\,x \to F\,x)$
   - $\forall x(F\,x \to H\,x), \exists x(G\,x \wedge \neg H\,x) \vdash \neg\forall x(F\,x \to G\,x)$
   - $\forall x\forall y\forall z(Rxy \wedge Ryz \to Rxz), \forall x\forall y(Rxy \to Ryx) \vdash \forall x\forall y(Rxy \to Rxx)$
   - $\exists x\exists y\, Rxy \vdash \exists x\exists y\, Ryx$
   - $\mathbf{T} : \forall x\forall y(Rxy \vee Ryx), \mathbf{T} : \exists x\exists y(Rxy \wedge \neg Ryx)$ (you may omit any branches that are identical to ones you develop)

5. Use the branching rules for first order logic to find a finite model that shows that $\forall x\exists y\, Rxy \vdash \exists x\forall y\, Rxy$ is not valid.