COMP2620/6262 (Logic) Tutorial

Week 5

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 **ten 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

Recall the natural deduction rules for first order logic:

$$\frac{}{\varphi \vdash \varphi}\ A$$

$$\frac{\Gamma \vdash \varphi \quad \Gamma' \vdash \psi}{\Gamma, \Gamma' \vdash \varphi \land \psi} \land I \qquad \frac{\Gamma \vdash \varphi \land \psi}{\Gamma \vdash \varphi} \land E1 \qquad \frac{\Gamma \vdash \varphi \land \psi}{\Gamma \vdash \psi} \land E2$$

$$\frac{\Gamma, \varphi \vdash \psi}{\Gamma \vdash \varphi \to \psi} \to I \qquad \frac{\Gamma \vdash \varphi \to \psi \quad \Gamma' \vdash \varphi}{\Gamma, \Gamma' \vdash \psi} \to E$$

$$\frac{\Gamma, \varphi \vdash \bot}{\Gamma \vdash \neg\varphi} \neg I \qquad \frac{\Gamma \vdash \neg\varphi \quad \Gamma' \vdash \varphi}{\Gamma, \Gamma' \vdash \bot} \neg E \qquad \frac{\Gamma \vdash \bot}{\Gamma \vdash \varphi} \bot E$$

$$\frac{\Gamma \vdash \varphi}{\Gamma \vdash \varphi \lor \psi} \lor I1 \qquad \frac{\Gamma \vdash \psi}{\Gamma \vdash \varphi \lor \psi} \lor I2 \qquad \frac{\Gamma \vdash \varphi \lor \psi \quad \Gamma', \varphi \vdash \sigma \quad \Gamma'', \psi \vdash \sigma}{\Gamma, \Gamma', \Gamma'' \vdash \sigma} \lor E$$

$$\frac{\Gamma \vdash \neg\neg\varphi}{\Gamma \vdash \varphi} \neg\neg E$$

$$\frac{}{\vdash t = t} = I \qquad \frac{\Gamma \vdash t = u \quad \Gamma' \vdash \varphi[t/x]}{\Gamma, \Gamma' \vdash \varphi[u/x]} = E$$

$$\frac{\Gamma \vdash \varphi[t/x]}{\Gamma \vdash \exists x\varphi} \exists I \qquad a \notin FV(\Gamma, \varphi, \Gamma', \psi) : \frac{\Gamma \vdash \exists x\varphi \quad \Gamma', \varphi[a/x] \vdash \psi}{\Gamma, \Gamma' \vdash \psi} \exists E$$

$$a \notin FV(\Gamma, \varphi) : \frac{\Gamma \vdash \varphi[a/x]}{\Gamma \vdash \forall x\varphi} \forall I \qquad \frac{\Gamma \vdash \forall x\varphi}{\Gamma \vdash \varphi[t/x]} \forall E$$

1. We will do our most complex modelling into formal language with the Logic4Fun tool, but here are some sentences to translate into first order logic notation, to help you get used to that notation.

   For these translations, assume that the universe of discourse is some collection of people. L is a unary predicate meaning 'is a librarian', R is a unary predicate meaning 'is a reader', N is a binary predicate meaning its first argument 'needs' its second argument, and f is a unary function mapping each person to their favourite person.

   - All librarians are readers.
     Solution. $\forall x(L\,x \to R\,x)$
   - Every reader needs a librarian.
     Solution. $\forall x(R\,x \to \exists y(L\,y \wedge N(x, y)))$
   - Some people's favourite person is themselves.
     Solution. $\exists x(f\,x = x)$
   - If anybody is a reader, then everybody needs librarians.
     Solution. $\exists x\,R\,x \to \forall y \exists z(L\,z \wedge N(y, z))$ - or does this sentence means that everyone needs *all* librarians, rather than just some of them?
   - Everybody needs their favourite person, but not everybody's favourite person needs them.
     Solution. $\forall x\,N(x, f\,x) \wedge \neg\forall y\,N(f\,y, y)$
   - If someone's favourite person is not a reader, then they must not be a librarian.
     Solution. The correct answer might be $\forall x(\neg R(f\,x) \to \neg L\,x)$, but also might be $\forall x(\neg R(f\,x) \to \neg L(f\,x))$; the dangling 'they' in the sentence is a source of ambiguity.

2. **The test at the start of the next tutorial will resemble this question. Natural deduction rules will be provided, as above. For full marks, you will need to use the five part notation taught in lectures: which premises are being used; a line number; a proposition; previous lines used; and the rule name.**

   Prove the following sequents by natural deduction.

   - $\forall x(Fx \to Gx),\ \forall x(Gx \to Hx)\ \vdash\ \forall x(Fx \to Hx)$
     Solution.

α_1	(1)	$\forall x(Fx \to Gx)$		A
α_2	(2)	$\forall x(Gx \to Hx)$		A
α_3	(3)	Fa		A
α_1	(4)	$Fa \to Ga$	1	$\forall$E
α_1, α_3	(5)	Ga	3, 4	$\to$E
α_2	(6)	$Ga \to Ha$	2	$\forall$E
$\alpha_1, \alpha_2, \alpha_3$	(7)	Ha	5, 6	$\to$E
α_1, α_2	(8)	$Fa \to Ha$	7	$\to$I
α_1, α_2	(9)	$\forall x(Fx \to Hx)$	8	$\forall$I

     The conclusion $\forall x(Fx \to Hx)$ is universal in form, so we aim to get it from a typical instance $Fa \to Ha$. This in turn is a conditional, so the usual $\to$I strategy applies: we assume its left (line 3) and derive its right (line 7). The overall shape of the proof is quite common in first order natural deduction: the quantifiers in the premises are stripped off so that we can do some propositional logic in the middle of the proof, and then the quantifiers are put back at the end.

   - $\forall x(\neg Fx \to Gx)\ \vdash\ \forall x(\neg Gx \to Fx)$
     Solution.

α_1	(1)	$\forall x(\neg Fx \to Gx)$		A
α_2	(2)	$\neg Ga$		A
α_3	(3)	$\neg Fa$		A
α_1	(4)	$\neg Fa \to Ga$	1	$\forall$E
α_1, α_3	(5)	Ga	$3, 4$	$\to$E
$\alpha_1, \alpha_2, \alpha_3$	(6)	$\bot$	$2, 5$	$\neg$E
α_1, α_2	(7)	$\neg\neg Fa$	6	$\neg$I
α_1, α_2	(8)	Fa	7	$\neg\neg$E
α_1	(9)	$\neg Ga \to Fa$	8	$\to$I
α_1	(10)	$\forall x(\neg Gx \to Fx)$	9	$\forall$I

Here the propositional logic in the middle of the proof involves the negation rules, including $\neg\neg$E. The strategy for handling the universal quantifier is essentially the same as in proof 1 above.

- $\forall x(Fx \to Gx),\ \exists x(Fx \land Hx)\ \vdash\ \exists x(Gx \land Hx)$

Solution.

α_1	(1)	$\forall x(Fx \to Gx)$		A
α_2	(2)	$\exists x(Fx \land Hx)$		A
α_3	(3)	$Fa \land Ha$		A
α_3	(4)	Fa	3	$\land$E
α_1	(5)	$Fa \to Ga$	1	$\forall$E
α_1, α_3	(6)	Ga	$4, 5$	$\to$E
α_3	(7)	Ha	3	$\land$E
α_1, α_3	(8)	$Ga \land Ha$	$6, 7$	$\land$I
α_1, α_3	(9)	$\exists x(Gx \land Hx)$	8	$\exists$I
α_1, α_2	(10)	$\exists x(Gx \land Hx)$	$2, 9$	$\exists$E

Note that the $\exists$I move has to be performed *before* the $\exists$E, so that the name a does not occur in the conclusion.

- $\forall x(Fx \to \neg Gx)\ \vdash\ \neg\exists x(Fx \land Gx)$

Solution.

α_1	(1)	$\forall x(Fx \to \neg Gx)$		A
α_2	(2)	$\exists x(Fx \land Gx)$		A
α_3	(3)	$Fa \land Ga$		A
α_3	(4)	Fa	3	$\land$E
α_1	(5)	$Fa \to \neg Ga$	1	$\forall$E
α_1, α_3	(6)	$\neg Ga$	$4, 5$	$\to$E
α_3	(7)	Ga	3	$\land$E
α_1, α_3	(8)	$\bot$	$6, 7$	$\neg$E
α_1, α_2	(9)	$\bot$	$2, 8$	$\exists$E
α_1	(10)	$\neg\exists x(Fx \land Gx)$	9	$\neg$I

- $\forall x(Fx \to Hx),\ \forall x(Gx \to \neg Hx)\ \vdash\ \neg\exists x(Fx \land Gx)$

Solution.

α_1	(1)	$\forall x(Fx \to Hx)$		A
α_2	(2)	$\forall x(Gx \to \neg Hx)$		A
α_3	(3)	$\exists x(Fx \wedge Gx)$		A
α_4	(4)	$Fa \wedge Ga$		A
α_4	(5)	Fa	4	$\wedge$E
α_1	(6)	$Fa \to Ha$	1	$\forall$E
α_1, α_4	(7)	Ha	5, 6	$\to$E
α_2	(8)	$Ga \to \neg Ha$	2	$\forall$E
α_4	(9)	Ga	4	$\wedge$E
α_2, α_4	(10)	$\neg Ha$	8, 9	$\to$E
$\alpha_1, \alpha_2, \alpha_4$	(11)	$\bot$	7, 10	$\neg$E
$\alpha_1, \alpha_2, \alpha_3$	(12)	$\bot$	3, 11	$\exists$E
α_1, α_2	(13)	$\neg\exists x(Fx \wedge Gx)$	12	$\neg$I

- $\neg\forall x(Fx \to Gx) \ \vdash \ \exists x(Fx \wedge \neg Gx)$

Solution.

α_1	(1)	$\neg\forall x(Fx \to Gx)$		A
α_2	(2)	$\neg\exists x(Fx \wedge \neg Gx)$		A
α_3	(3)	Fa		A
α_4	(4)	$\neg Ga$		A
α_3, α_4	(5)	$Fa \wedge \neg Ga$	3, 4	$\wedge$I
α_3, α_4	(6)	$\exists x(Fx \wedge \neg Gx)$	5	$\exists$I
$\alpha_2, \alpha_3, \alpha_4$	(7)	$\bot$	2, 6	$\neg$E
α_2, α_3	(8)	$\neg\neg Ga$	7	$\neg$I
α_2, α_3	(9)	Ga	8	$\neg\neg$E
α_2	(10)	$Fa \to Ga$	9	$\to$I
α_2	(11)	$\forall x(Fx \to Gx)$	10	$\forall$I
α_1, α_2	(12)	$\bot$	1, 11	$\neg$E
α_1	(13)	$\neg\neg\exists x(Fx \wedge \neg Gx)$	12	$\neg$I
α_1	(14)	$\exists x(Fx \wedge \neg Gx)$	13	$\neg\neg$E

This cannot be proved without classical reasoning. Lines 2 – 10 constitute a proof of the contraposed version

$\neg\exists x(Fx \wedge \neg Gx) \ \vdash \ \forall x(Fx \to Gx)$

which is easier to approach as it falls to the usual $\to$I strategy.

- $\vdash \ \exists x(Fx \to \forall y Fy)$ (Tricky)

Solution.

α_1	(1)	$\neg\exists x(Fx \to \forall y Fy)$		A
α_2	(2)	$\neg Fa$		A
α_3	(3)	Fa		A
α_2, α_3	(4)	$\bot$	2, 3	$\neg$E
α_2, α_3	(5)	$\forall y Fy$	4	$\bot$E
α_2	(6)	$Fa \to \forall y Fy$	5	$\to$I
α_2	(7)	$\exists x(Fx \to \forall y Fy)$	6	$\exists$I
α_1, α_2	(8)	$\bot$	1, 7	$\neg$E
α_1	(9)	$\neg\neg Fa$	8	$\neg$I
α_1	(10)	Fa	9	$\neg\neg$E
α_1, α_3	(11)	$\forall y Fy$	10	$\forall$I
α_1	(12)	$Fa \to \forall y Fy$	11	$\to$I
α_1	(13)	$\exists x(Fx \to \forall y Fy)$	12	$\exists$I
α_1	(14)	$\bot$	1, 13	$\neg$E
	(15)	$\neg\neg\exists x(Fx \to \forall y Fy)$	14	$\neg$I
	(16)	$\exists x(Fx \to \forall y Fy)$	15	$\neg\neg$E

This is perhaps the strangest theorem proved in the entire course. The sequent is sometimes called the "key drinker theorem", as an instance of it states: "There is someone such that if they drinks, then everybody drinks!" Its proof requires all the resources of classical logic.

Note in particular the unusual decision to weaken on line 11 to put α_3 back into our premises, because we wished to use that premise again.

3. Attempt the following puzzles from the Logic4Fun website.

- Office Blocked

  Solution.
  Constraints:
  ```
  ALL x ALL y (x <> y IMP floor(x) <> floor (y) OR number(x) <> number(y)).
  floor A = 2 AND floor B = 1 AND number A = number B.
  floor D = 1 AND (number B) DIF (number D) = 1.
  number E < number F.
  floor C = floor F AND (number C) DIF (number F) = 1.
  ```
- Library Books

  Solution.
  Sorts:
  ```
  student enum : Arthur, Belinda, Clarissa.
  subject enum : English, French, German.
  ```
  Vocabulary:
  ```
  function {
  books (student) : natnum.
  studiedBy (subject) : student {all_different}.
  }
  ```
  Constraints:
  ```
  books Arthur + books Belinda + books Clarissa = 25.
  ALL x (books x < 15).
  books (studiedBy French) = books Arthur + 3.
  books Clarissa = books (studiedBy English) + 2.
  ```
- Lost Property

  Solution.
  Sorts:
  ```
  person enum : Alex, Coco, Dodder, Ferdinand, Flo.
  item enum: briefcase, cheese, redNose, teddyBear, umbrella.
  day enum : Monday, Tuesday, Wednesday, Thursday, Friday.
  shop enum : cafe, newsagency, pharmacy, shoeshop, supermarket.
  ```
  Vocabulary:
  ```
  function {
  what (person) : item {all_different}.
  when (item) : day {all_different}.
  where (day) : shop {all_different}.
  }
  ```
  Constraints:
  ```
  what Coco = redNose AND where (PRED (when redNose)) = pharmacy.
  where (when umbrella) = newsagency.
  where Monday = cafe.
  where (when (what Dodder)) = shoeshop.
  when (what Ferdinand) = Wednesday.
  when briefcase = Tuesday AND when (what Flo) > Tuesday.
  where (SUCC (when (teddyBear))) = supermarket AND when teddyBear <> Thursday.
  ```
- Close Encounter

  Solution.
  Sorts:
  ```
  alien enum : V, W, X, Y, Z.
  planet enum : A, B, C, D, E.
  drink enum : blackHole, nebula, whiteDwarf, warpFive, supernova.
  job enum : prospector, mercenary, gambler, roboticist, generalist.
  ```
  Vocabulary:

```
function {
from (alien) :  planet { all_different } .
drinks (alien) :  drink { all_different } .
does (alien) :  job { all_different } .
}
```

Constraints:

```
from V = C.
does X = roboticist AND SOME u (does u = generalist AND from u = E).
drinks Z <> whiteDwarf.
drinks Y = supernova AND SOME u (drinks u = warpFive AND from Y = PRED (from u)).
SOME u (does u = prospector AND from u <> D AND from u <> B).
SOME u (from u = B AND drinks u = blackHole).
SOME u (does u = gambler AND from u = A AND drinks u <> nebula).
```

We could have approached this question as we did Lost Property, making sure every function composes to make it easy to compare one sort with another. Instead this solution makes the alien sort the domain of every function, and uses the `SOME` quantifier to relate other sorts. As another alternative, we could have used `ALL` instead of `SOME`, for example writing

```
ALL u (does u = generalist IMP from u = E).
```