COMP2620/6262 (Logic) Tutorial

Week 10

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.

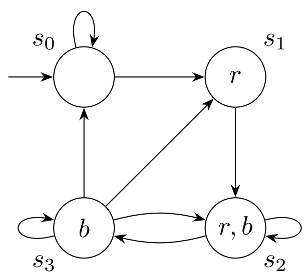
This week's quiz is on **tableaux** for LTL. Your tutor will hand out blank paper, on which you should clearly write your university ID and name. Your tutor will also hand out paper with all tableaux rules for LTL, including loop and simple repetition rules. They will then write two signed propositions on the whiteboard. You should then construct a **complete** tableaux for those signed propositions. In other words, if a branch terminates, you should cross if appropriate, indicate with [LOOP,n] or [REP,n] for some number n if applicable, then move on to complete all non-terminated branches. You will not need to extract a satisfying model. You will have **twenty minutes** to do this.

If your tableau becomes repetitive due to similar or identical propositions appearing in multiple nodes, you may omit some steps with a short English explanation of the reason. Do not skip any steps, including numbering and justifying your lines, the first time you deal with any propositions.

You are not permitted to have any other resource on the table during this quiz, including any electronic device. If you finish your quiz before time elapses you may put your hand up and your tutor will collect your sheet. Once you have done this, you may get a device out and start work silently on this week's questions. If you are still working when time elapses you must stop writing immediately and let your tutor collect your paper.

This Week's Exercises

1. Consider a transition system representing a machine that can receive one or more unstarted requests (r), and become busy fulfilling these requests (b). Here the notion of 'next state' will correspond to a real clock ticking, and therefore the machine might, for example, stay busy for an unpredictable number of steps, and might receive new requests while it is busy. It also has to become busy before it is able to label a request as started (which will negate r, unless new requests come in the meantime).



   Draw the corresponding computation tree up to level 4, where the root counts as level 0.

2. State the following CTL propositions in plain and intuitive English, and discuss informally whether or not they hold for the machine of the previous question.

   - EGb
   - $AG(r \to AXb)$

- $EF(b \wedge \neg r \wedge EX(\neg b \wedge r))$
- $AGAF(r \to b)$
- $E[\neg r\, U\, b]$
- $E[\neg r\, U\, AXAXb]$

3. Are the following equivalences of CTL? If so, argue why, using the equivalences learned in class. If not, define a transition system that affirms one and negates the other.

   - $\neg(AFp \vee EG\neg q)$ and $EG\neg p \wedge AFq$
   - $AXEXp$ and $EXAXp$.
   - $\neg AGEGp$ and $AFEF\neg p$
   - $\neg A[p\, U\, q]$ and $AFq \to E[\neg p\, U\, \neg(p \vee q)]$

4. The LTL tableaux rule for $\mathbf{T} : F\varphi$ branches into two possibilities: $\mathbf{T} : \varphi$ and $\mathbf{T} : XF\varphi$. Justify this with a mathematical agument (not a tableaux proof!) using the LTL semantics, that $F\varphi$ and $\varphi \vee XF\varphi$ are equivalent.

   Could a similar splitting into a disjunction be possible for AF and/or EF in CTL? Again, use the semantics as justification.

5. **The test at the start of the next tutorial will resemble this question, by asking you to model English language sentences into CTL*.**

   As an aspiring logician, you are often approached by people demanding that you formalise their opinions and ravings. Today's customer is a wizard who wants you to record their observations and predictions about the world of magic and the supernatural. Luckily, although you are not exactly an expert in this domain, you can see that CTL* can express their statements. Define some appropriate propositional variables, then formalise the following:

   - "It is necessary that eventually the phoenix shall burn"
   - "It may be that the fairies shall always fail to fly"
   - "It must be either that at some time the phoenix shall burn, or that tomorrow the goblins shall prosper"
   - "Perhaps it is true both that goblins will prosper until the sirens sing, and that if the fairies are flying then the sirens shall sing forever"

   Bad news! After the wizard occupied your time, a rugby fanatic has tackled you and demanded a logical translation of their outlandish claims about prospects for the current and future seasons of four Australian rugby union teams. These you must also translate into CTL*:

   - "Possibly, the Waratahs will win forever"
   - "We will definitely eventually have a season where both the Reds and Force win"
   - "Maybe, at some time, the Reds will not win in their next season"
   - "Certainly, either the Brumbies are winning, or the Brumbies will win until it becomes necessary that the Reds win forever"

6. (Tricky and Open Ended) How might 'past operators' work in LTL, CTL, or CTL*? By past operators we mean for example: X^{-1}, a unary connective meaning 'on the immediately previous state in the path considered'; F^{-1}, meaning 'there exists a previous state in the path considered', $(AF)^{-1}$, meaning 'for all paths that lead to the current state, there exists a previous state in such a path', and so on. Would adding these operators to our logics make them more expressive, or can we express them using the logics we already have?