Informatics 1 - Computation & Logic: Tutorial 9

Revision

Week 11: 25 - 29 November 2013

Please attempt the entire worksheet in advance of the tutorial, and bring with you all work, including (if a computer is involved) printouts of code and test results. Tutorials cannot function properly unless you do the work in advance.

You may work with others, but you must understand the work; you can't phone a friend during the exam.

Assessment is formative, meaning that marks from coursework do not contribute to the final mark. But coursework is not optional. If you do not do the coursework you are unlikely to pass the exams.

Attendance at tutorials is **obligatory**; please let your tutor know if you cannot attend.

Truth tables

1. Construct a truth table for the following expression of propositional logic, and use these to decide whether the expressions are contingent, tautologous or inconsistent:

   $\neg(A \wedge B) \wedge (B \vee (C \rightarrow \neg A))$

This expression is CONTINGENT/TAUTOLOGOUS/INCONSISTENT

Sequent Calculus

2. Assume the following proof rules from Tutorial Exercise 4, known respectively as '*immediate*', '$\wedge$ *introduction*', '$\rightarrow$ *introduction*', '$\wedge$ *elimination*' and '$\rightarrow$ *elimination*':

$$\frac{\mathcal{A}, X \vdash X}{} \qquad \frac{\mathcal{A} \vdash X \wedge Y}{\begin{matrix}\mathcal{A} \vdash X \\ \mathcal{A} \vdash Y\end{matrix}} \qquad \frac{\mathcal{A} \vdash X \rightarrow Y}{\mathcal{A}, X \vdash Y}$$

$$\frac{\mathcal{A}, X \wedge Y \vdash Z}{\mathcal{A}, X, Y \vdash Z} \qquad \frac{\mathcal{A}, X \rightarrow Z \vdash Z}{\mathcal{A} \vdash X}$$

Prove that the following argument is valid using this method:

$[\mathsf{A}, (\mathsf{A} \wedge \mathsf{B}) \rightarrow \mathsf{C}] \vdash \mathsf{B} \rightarrow \mathsf{C}$

3. Here are some more proof rules, called respectively '$\vee$*introduction left*', '$\vee$*introduction right*', and '$\vee$*elimination':*

$$\frac{\mathcal{A} \vdash X \vee Y}{\mathcal{A} \vdash X} \qquad \frac{\mathcal{A} \vdash X \vee Y}{\mathcal{A} \vdash Y} \qquad \frac{\mathcal{A},\, X \vee Y \vdash Z}{\begin{array}{c}\mathcal{A}, X \vdash Z \\ \mathcal{A}, Y \vdash Z\end{array}}$$

Prove that the following argument is valid:

$[(\mathsf{A} \vee \mathsf{B}) \rightarrow \mathsf{C},\, \mathsf{C} \rightarrow \mathsf{A}] \vdash \mathsf{B} \rightarrow \mathsf{C}$

Resolution

4. Using resolution, prove whether the following argument is valid:

$(P \wedge Q) \to R, \neg R \vdash P \to \neg Q$

(a) Convert into conjunctions:

(b) Convert to conjunctive normal form:

(c) Convert into clausal form:

(d) Apply the resolution rule:

5. Using resolution, prove whether the following argument is valid:
$P \to Q, P \to R \vdash P \to (Q \wedge R)$

   (a) Convert into conjunctions:

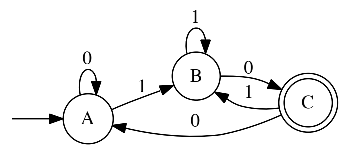
   (b) Convert to conjunctive normal form:

   (c) Convert into clausal form:

   (d) Apply the resolution rule:

Finite State Machines

6. Consider the following finite state acceptor over the alphabet $\{0, 1\}$:



   (a) Draw the transition table for this machine:

   (b) For each of the following strings, say whether they are *accepted* or *rejected* by this machine and indicate in which state the machine ends. Note that ϵ denotes the empty string.

STRING	ACCEPT/REJECT	FINAL STATE
110		
1010		
101		
0011010		
10011		
110110		
0110		
00		
00111		
ϵ		

   (c) Give the language for the above FSM:

7. Given the following languages $L_1(0(10)^*)$, $L_2((1|0^*)1)$ and $L_3((11)^*0)$,

   (a) Draw deterministic or non-deterministic machines M_i for each of the languages L_i, such that:

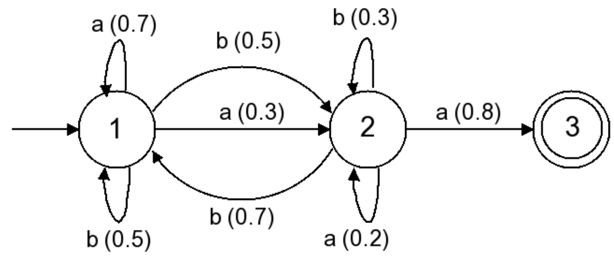
      i. $L(M_1) = L_1$

      ii. $L(M_2) = L_2$

      iii. $L(M_3) = L_3$

   (b) Compose the machines together to create a machine, M, such that $L(M) = (L(M_1)|L(M_2))(L(M_3))^*$

8. Consider the following Probabilistic Finite State Machine that accepts strings over the alphabet `{a, b}`.



(a) Identify all possible traces that would accept string `abaa`.

(b) Calculate the probability of each trace identified.

(c) Finally, calculate the probability that the string `abaa` would be accepted by this FSM.

This tutorial exercise sheet was originally written by Paolo Besana and extended by Thomas French and Areti Manataki. Send comments to `A.Manataki@ed.ac.uk`