Informatics 2A: Tutorial Sheet 9 (Week 11)
Revision Tutorial
Sample 'Part A' exam questions

Mary Cryan, Shay Cohen

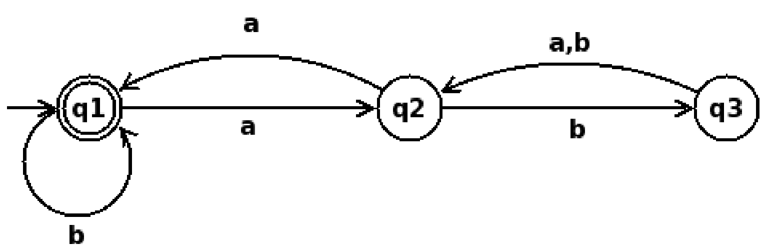
The Informatics 2A exam this year will have two parts. Part A will consist of five compulsory short questions, worth 10 marks each. Part B will consist of three long questions, of which you must attempt two; these are worth 25 marks each. As a guideline, allowing for time taken to familiarize yourself with the paper as a whole, you should spend 10 minutes on each of the Part A questions.

This tutorial sheet consists of six sample problems illustrating the style and length of Part A questions. (Naturally, nothing is implied about the selection of topics that will feature in the exam.) The guideline time for completing this tutorial sheet under exam conditions is therefore 1 hour.

Examples of Part B style questions can be found in Informatics 2A exam papers from recent years; these are available from

`http://www.inf.ed.ac.uk/teaching/courses/inf2a/past_exams/`

1. Consider the DFA below.



   (a) Based on the DFA, write down simultaneous Kleene-algebra equations in three variables X_1, X_2 and X_3, representing the languages accepted if the DFA is run starting in states $q1$, $q2$ and $q3$ respectively. [*4 marks*]

   (b) Use Arden's Rule to solve the simultaneous equations and produce a regular expression for the language accepted by the DFA (whose start state is $q1$). [*6 marks*]

2. Consider a pushdown automaton with two control states $Q = \{q1, q2\}$, start state $q1$, input alphabet $\Sigma = \{a, b\}$, stack alphabet $\Gamma = \{a, b, \bot\}$, and transition relation:

$$\begin{array}{ccccccc} q1 & \xrightarrow{a,\,x\,:\,ax} & q1 & \qquad & q2 & \xrightarrow{a,\,a\,:\,\epsilon} & q2 \\ q1 & \xrightarrow{b,\,x\,:\,bx} & q1 & & q2 & \xrightarrow{b,\,b\,:\,\epsilon} & q2 \\ q1 & \xrightarrow{\epsilon,\,x\,:\,x} & q2 & & q2 & \xrightarrow{\epsilon,\,\bot\,:\,\epsilon} & q2 \end{array}$$

where x can be instantiated with any symbol from the stack alphabet Γ. The automaton accepts on empty stack.

(We use the general notation

$$q \xrightarrow{s\,x\,:\,\alpha} q'$$

to mean that when the automaton is in control state $q \in Q$ and $x \in \Gamma$ is popped from the top of the stack, the input symbol or empty string $s \in \Sigma \cup \{\epsilon\}$ can be read to reach control state $q' \in Q$ with $\alpha \in \Gamma^*$ pushed onto the stack.)

(a) Describe in detail an execution of the above PDA that accepts the string

$$abba$$

[*8 marks*]

(b) Give a precise mathematical definition of the language recognised by the PDA above. [*2 marks*]

3. (a) For each of the following languages, state the lowest level of the Chomsky hierarchy at which the language resides. You need not justify your answers: [*4 marks*]

   i. $\{a^\ell b^m c^n \mid \ell, m, n \in \mathbb{N}\}$
   ii. $\{a^m b^m c^n \mid m, n \in \mathbb{N}\}$
   iii. $\{a^n b^n c^n \mid n \in \mathbb{N}\}$
   iv. $\{ww \mid w \in \{a, b, c\}*\}$

(b) The *context-free pumping lemma* is a useful tool for showing that certain languages are context-free. It states that if L is a context-free language, there exists $k > 0$ such that any string $s \in L$ of length at least k may be written as $uvwxy$ where $|vwx| \leq k$, $vx \neq \epsilon$, such that for any i we also have $uv^iwx^iy \in L$.

Explain informally why the context-free pumping lemma is true, possibly with the help of diagrams. Your explanation does not have to be a precise mathematical proof, but it should be intuitively convincing at least. [*6 marks*]

(for this question, you are not expected to apply the lemma to any of the languages from part (a)).

4. Consider the following probabilistic context-free grammar:

$$
\begin{array}{lcl}
\mathsf{S} & \rightarrow & \mathsf{NP}\ \mathsf{VP}\ \ (1.0) \\
\mathsf{NP} & \rightarrow & \mathsf{N}\ \ (0.5)\ \ |\ \mathsf{A}\ \ \mathsf{NP}\ \ (0.5) \\
\mathsf{VP} & \rightarrow & \mathsf{V}\ \ (0.5)\ \ |\ \mathsf{V}\ \ \mathsf{NP}\ \ (0.5) \\
\mathsf{N} & \rightarrow & \textit{fat}\ (0.2)\ \ |\ \textit{ducks}\ (0.4)\ \ |\ \textit{fish}\ (0.4) \\
\mathsf{V} & \rightarrow & \textit{ducks}\ (0.4)\ \ |\ \textit{fish}\ (0.6) \\
\mathsf{A} & \rightarrow & \textit{fat}\ (1.0)
\end{array}
$$

(a) Draw up a probabilistic CYK chart (in matrix form) for the sentence

fat ducks fish

Include all entries, whether or not they contribute to some complete parse. [*8 marks*]

(b) Draw the most probable parse tree for the sentence. What is its probability? [*2 marks*]

5. Consider the following grammar with associated semantics:

$$
\begin{array}{rll}
\mathsf{S} \rightarrow & \mathsf{PN}\ \text{is a}\ \mathsf{NP} & \{\ \mathsf{NP.Sem(PN.Sem)}\ \} \\
\mathsf{S} \rightarrow & \mathsf{PN}\ \text{is not a}\ \mathsf{NP} & \{\ \neg\mathsf{NP.Sem(PN.Sem)}\ \} \\
\mathsf{NP} \rightarrow & \mathsf{N}\ \mathsf{PrP} & \{\ \lambda \text{x}.\mathsf{N.Sem}(\text{x}) \wedge \mathsf{PrP.Sem}(\text{x})\ \} \\
\mathsf{PrP} \rightarrow & \mathsf{Prep}\ \mathsf{PN} & \{\ \mathsf{Prep.Sem(PN.Sem)}\ \} \\
\mathsf{PN} \rightarrow & \textit{London} & \{\ \text{London}\ \} \\
\mathsf{PN} \rightarrow & \textit{Scotland} & \{\ \text{Scotland}\ \} \\
\mathsf{N} \rightarrow & \textit{city} & \{\ \lambda \text{z. city(z)}\ \} \\
\mathsf{N} \rightarrow & \textit{country} & \{\ \lambda \text{z. country(z)}\ \} \\
\mathsf{Prep} \rightarrow & \textit{in} & \{\ \lambda \text{y}.\lambda \text{z. In(z,y)}\ \} \\
\mathsf{Prep} \rightarrow & \textit{outside} & \{\ \lambda \text{y}.\lambda \text{z.}\ \neg\text{In(z,y)}\ \}
\end{array}
$$

(a) Compute the semantics of the following sentence. Show your working in detail, including any relevant β-reduction steps.

London is a city outside Scotland

[*7 marks*]

(b) Write down the logical formula obtained as the interpretation of

London is not a city in Scotland

You need not show your working in this case. [*1 mark*]

(c) Look at the logical formulae obtained from the two complete sentences above. Does either imply the other via the usual laws of propositional logic? In which direction(s) does the implication hold? [*2 marks*]

6. Let Σ be a set of terminals consisting of all words in the English language together with the exclamation mark '!'. We consider a grammar with nonterminals

$$\mathsf{S} \quad \mathsf{E} \quad \mathsf{A}_x$$

where x ranges over words in the English language; that is, there is one nonterminal A_x, for every English word x. The grammar contains the following rules.

$$\begin{array}{rcl} \mathsf{S} & \to & \mathsf{A}_x \ \mathsf{E} \ \mid \ \mathsf{A}_x \ \mathsf{S} \ \mathsf{E} \\ \mathsf{A}_x \ \mathsf{E} & \to & x \ ! \\ \mathsf{A}_x \ y & \to & y \ \mathsf{A}_x \\ y \ \mathsf{A}_x \ ! & \to & y \ y \ ! \ \mathsf{A}_x \end{array}$$

where x, y again range over English words. The start symbol is S.

(a) Give a full derivation of

$$\mathsf{hip\ hip\ !\ hooray\ !}$$

[*6 marks*]

(b) Give a precise mathematical description of the language L generated by the grammar. [*2 marks*]

(c) State where the language l resides in the Chomsky hierarchy, and give brief justification for why the language does indeed reside at the level you assert. (You do not need to explain why the language does not reside lower down the hierarchy.) [*2 marks*]