UNIVERSITY OF EDINBURGH

COLLEGE OF SCIENCE AND ENGINEERING

SCHOOL OF INFORMATICS

INFR08008 INFORMATICS 2A: PROCESSING FORMAL AND NATURAL LANGUAGES

Saturday 12th December 2015

09:30 to 11:30

INSTRUCTIONS TO CANDIDATES

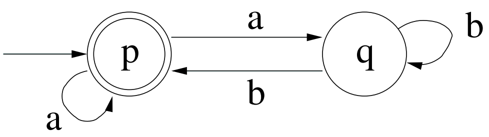
1. **Answer all five questions in Part A, and two out of three questions in Part B. Each question in Part A is worth 10% of the total exam mark; each question in Part B is worth 25%.**
2. **Use a single script book for all questions.**
3. **Calculators may be used in this exam.**

Convener: D. K. Arvind
External Examiner: C. Johnson

THIS EXAMINATION WILL BE MARKED ANONYMOUSLY

Part A

1. Consider the following NFA with alphabet $\{a, b\}$:



   (a) Draw the state diagram for the equivalent DFA obtained via the standard construction. [*4 marks*]

   (b) Use your DFA from part (a) to derive a regular expression for the language it accepts. Show your working. Give your regular expression in mathematical notation. [*4 marks*]

   (c) Now write down the same regular expression in machine syntax (as used by the `grep` tool, for example). [*2 marks*]

2. Consider the following simple grammar for Unix-style shell commands. The terminals are *command*, *option*, *file*; the start symbol is the non-terminal shell.

$$\begin{array}{rcl} \textsf{shell} & \rightarrow & \textit{command}\ \textsf{args} \\ \textsf{args} & \rightarrow & \textsf{opts files} \\ \textsf{opts} & \rightarrow & \epsilon \mid \textit{option}\ \textsf{opts} \\ \textsf{files} & \rightarrow & \epsilon \mid \textit{file}\ \textsf{files} \end{array}$$

   (a) Write down the set E of potentially empty non-terminals, and calculate the First and Follow sets for all the non-terminals in the grammar. [*5 marks*]

   (b) Using this information, or by simple inspection, draw up the LL(1) parse table for the grammar. [*5 marks*]

3. Consider the following hidden Markov model tables:

   Emission matrix:

	keys	open	gates
N	0.4	0.2	0.4
V	0.05	0.7	0.05

   Transition matrix (rows denote the current part of speech, and columns denote the part of speech we transition to):

	N	V
$\langle s \rangle$	0.7	0.1
N	0.1	0.1
V	0.5	0.1

   (a) Are these matrices valid hidden Markov tables? If not, explain what is wrong with them. *[2 marks]*

   (b) Consider the sentence *[6 marks]*

   ```
   Keys open gates
   ```

   Apply the Viterbi algorithm on it and find the best POS sequence assuming the above hidden Markov model tables.

   (c) Show how to obtain a (non-normalized) probabilistic context-free grammar that attaches the same probability to every sentence as the above HMM does. You need not give all the rules and probabilities, but should give a representative sample in order to illustrate the idea. *[2 marks]*

4. (a) Give an English sentence which is ambiguous because of ambiguity in the part-of-speech tags. Describe two part-of-speech analyses for it. *[3 marks]*

   (b) Give a sentence involving structural ambiguity that does not originate from the part-of-speech tags. Indicate two possible syntactic analyses for it (you need not draw complete parse trees). *[2 marks]*

   (c) Consider the sentence

   ```
   England borders Scotland
   ```

   Suggest a context-free grammar that generates this sentence. The grammar should use the nonterminals `S`, `NP`, `V` (for verb), and `N` (for proper noun). Then augment your grammar with suitable semantic attachments such that the above sentence will yield the the lambda expression

   ```
   borders(England,Scotland)
   ```

   as its semantics. (You need not display the working that derives this lambda expression from a parse tree for the sentence.) *[5 marks]*

5. (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.

   i. $\{a^l b^m c^n \mid l, m, n \in \mathbb{N}\}$

   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. $\{ww \mid w \in \{a, b, c\}^*\}$

   [*4 marks*]

   (b) The *context-free pumping lemma* is a useful tool for showing that certain languages are not 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 $|vwx| \leq k$, $vx \neq \epsilon$, and 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 need not amount to a precise mathematical proof, but should be intuitively convincing. You are not expected to apply the lemma to any of the languages from part (a). [*6 marks*]

Part B

6. In this question, we will follow a proof that natural languages are not regular. We break the proof down into simple steps.

   (a) Consider the set of nouns *cat*, *dog*, *rat*, *elephant* (N), the determiner *the* (D), the transitive verbs *bit*, *chased* (VT), and the intransitive verb *died* (VI). Write down a context-free grammar that generates grammatical sentences of the form

   $$(\mathsf{D}\ \mathsf{N})^n\ \mathsf{VT}^{n-1}\ \mathsf{VI}$$

   (and only these sentences). [*6 marks*]

   (b) Prove using the pumping lemma that the language consisting of sentences $(\mathsf{D}\ \mathsf{N})^n\ \mathsf{VT}^{n-1}\ \mathsf{VI}$ is not regular. [*8 marks*]

   (c) Define a *regular* language such that its intersection with the set of grammatical English sentences yields precisely the language considered in part (b). Explain briefly why your language has this property. Use this to prove that English itself is *not* regular. [*7 marks*]

   (d) Are context-free grammars sufficient for describing the structure of *all* natural languages? Briefly justify your answer. [*4 marks*]

7. Consider the grammar

$S \rightarrow$ NP VP | NP VPP
VPP $\rightarrow$ VP PP
VP $\rightarrow$ V NPP | V NP
NPP $\rightarrow$ NP PP
PP $\rightarrow$ Prep NP
NP $\rightarrow$ Det N | Adj N | Det Adj N
V $\rightarrow$ bit | chased
Adj $\rightarrow$ shiny | white | big
Det $\rightarrow$ the | a
N $\rightarrow$ dog | boy | girl | house | teeth
Prep $\rightarrow$ with

(a) Consider the sentence *[8 marks]*

`The dog bit the boy with shiny teeth.`

Construct a CKY chart for this sentence in upper matrix form, and specify in each relevant cell the set of nonterminals for the corresponding substring that the CKY algorithm will discover.

(b) The CKY algorithm will find multiple parse trees for the above sentence. Draw all parse trees for the complete sentence as discovered by the algorithm. *[5 marks]*

(c) Give an informal explanation of the different meanings corresponding to each of the structures drawn in part (b). *[5 marks]*

(d) Choose the parse tree in which the non-terminal VPP dominates the most words. Allocate probabilities to each of the grammar rules such that the probabilistic version of CKY will yield that parse tree (note that you have to allocate probabilities so as to yield a valid PCFG). You may use the probability 0 for certain rules if you like. Write down the probability of this parse tree according to your PCFG. *[5 marks]*

(e) Consider the sentence *[2 marks]*

`The boy bit the dog with shiny teeth.`

Suppose we use probabilistic CKY to parse this sentence with the PCFG created in part (d). What is the probability of the parse tree probabilistic CKY finds? Briefly explain the reason for the relationship between this probability and the one from (d).

8. Consider the non-deterministic pushdown automaton with four states q_0, q_1, q_2, q_3, input alphabet $\{a, b\}$, stack alphabet $\{\perp, *\}$, and the transitions displayed below. Here x may stand for either a or b, and s may stand for either $\perp$ or $*$, so that lines 1,2,3,5,6 and 7 below strictly speaking represent two transitions each.

$$\begin{array}{ccc}
q_0 & \xrightarrow{x,\perp\,:\,*\perp} & q_0 \\
q_0 & \xrightarrow{a,s\,:\,s} & q_1 \\
q_1 & \xrightarrow{x,*\,:\,\epsilon} & q_1 \\
q_1 & \xrightarrow{\epsilon,\perp\,:\,\perp} & q_2 \\
q_2 & \xrightarrow{x,\perp\,:\,*\perp} & q_2 \\
q_2 & \xrightarrow{b,s\,:\,s} & q_3 \\
q_3 & \xrightarrow{x,*\,:\,\epsilon} & q_3 \\
q_3 & \xrightarrow{\epsilon,\perp\,:\,\epsilon} & q_3
\end{array}$$

The start state is q_0, and the initial stack state is $\perp$. Acceptance is by empty stack. Notice especially that the transition $q_0 \to q_1$ can be made only on input a, and the transition $q_2 \to q_3$ only on input b.

(a) Study the above transitions in order to understand how this machine behaves. Then draw up a table displaying the execution of this machine on the input string

$$aabaabba$$

(Hint: first work out at which point in the string the ϵ-transition $q_1 \to q_2$ will need to be made, by considering the various possibilities.) [*10 marks*]

(b) Suppose s is a string of length $2n$ that is accepted by our PDA. Suppose moreover that the transition $q_0 \to q_1$ is made on an occurrence of a as the ith symbol of s. What may be said about the $(i+n)$th symbol of s? Justify your answer. [*3 marks*]

(c) Why can our PDA not accept any strings of the form ww where $w \in \{a, b\}^*$? [*2 marks*]

(d) Design a context-free grammar for the language accepted by the above PDA. The terminal alphabet will be $\{a, b\}$, but you may introduce whatever non-terminals you need. You are advised to create your grammar directly from your understanding of the PDA, rather than by trying to follow some general algorithm. [*6 marks*]

(e) Sketch how you would modify your solution to part (d) to obtain a grammar that generates precisely those strings $s \in \{a, b\}^*$ that are *not* of the form ww. Justify your answer. [*4 marks*]