UNIVERSITY OF EDINBURGH

COLLEGE OF SCIENCE AND ENGINEERING

SCHOOL OF INFORMATICS

INFORMATICS 2A: PROCESSING FORMAL AND NATURAL LANGUAGES

Monday 13 th August 2012

14:30 to 16:30

Convener: J Bradfield
External Examiner: A Preece

INSTRUCTIONS TO CANDIDATES

1. **Answer Parts A and B. The multiple choice questions in Part A are worth 50% in total and are each worth the same amount. Mark one answer only for each question — multiple answers will score 0. Marks will not be deducted for incorrect answers. Part B contains THREE questions. Answer any TWO. Each is worth 25%.**
2. **Use the special mark sheet for Part A. Use a separate script book for each of the two questions from Part B that you answer.**

PART A
ANSWER ALL QUESTIONS IN PART A. Use the special mark sheet.

1. Which of the following strings is a member of the language over $\{a, b\}$ defined by the regular expression $(aa + ba)^*(bb)^*$?

   (a) *aabbba*

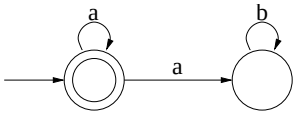
   (b) *aaaabb*

   (c) *babbaa*

   (d) *bbaa*

   (e) None of the above.

2. Suppose we apply the *subset construction* to the following NFA over $\{a, b\}$:



   Ignoring unreachable states, the resulting DFA has:

   (a) 2 states, 1 of which is accepting

   (b) 3 states, 1 of which is accepting

   (c) 3 states, 2 of which are accepting

   (d) 4 states, 1 of which is accepting

   (e) 4 states, 2 of which are accepting.

3. Consider the language L over $\{a, b\}$ consisting of strings with twice as many a's as b's. You are using the pumping lemma to show that L is not regular. You begin by supposing L were accepted by a DFA with k states. Following the recipe given in lectures for applying the pumping lemma, what would be a good choice of a string xyz that would allow you to obtain a contradiction?

   (a) *aaaabb*

   (b) *aaabbb*

   (c) $a^k b^{2k}$

   (d) $a^k b^k$

   (e) None of the above.

4. Consider the following context-free grammar, with start symbol S and terminals `a`, ;, <, >.

$$S \rightarrow <L \mid a \qquad L \rightarrow \texttt{a}R \mid <LR \qquad R \rightarrow > \mid ;L$$

How many different parse trees are there for the string $<<\texttt{a}>;\texttt{a}>$?

(a) 0

(b) 1

(c) 2

(d) 3

(e) More than three.

5. Which one of the following grammars is in *Chomsky normal form*? In each grammar, terminals are in lowercase, nonterminals are in uppercase, and the start symbol is S.

(a) $X \rightarrow YZ, \quad Y \rightarrow a, \quad Z \rightarrow bc$

(b) $X \rightarrow aY, \quad Y \rightarrow b \mid c$

(c) $X \rightarrow YZW, \quad Y \rightarrow a, \quad Z \rightarrow b, \quad W \rightarrow c$

(d) $X \rightarrow YZ, \quad Y \rightarrow WW, \quad Z \rightarrow a, \quad W \rightarrow b \mid c$

(e) $X \rightarrow YY \mid c, \quad Y \rightarrow Z, \quad Z \rightarrow a \mid b$

6. Which one of the following statements concerning pushdown automata is *false*? (Here we use 'accept' to mean 'accept on empty stack'.)

(a) Every context-free language is accepted by some deterministic PDA.

(b) The language accepted by any non-deterministic PDA is a context-free language.

(c) For any non-deterministic PDA, there is a non-deterministic PDA with just one state that accepts the same language.

(d) Every LL(1) parse table gives rise to a deterministic PDA.

(e) There are context-sensitive languages that are not accepted by any non-deterministic PDA.

7. Which of the following rules would not be permitted in a context-sensitive grammar? Again, terminals are in lowercase and nonterminals in uppercase.

(a) $A \rightarrow bAc$

(b) $bA \rightarrow aB$

(c) $Ab \rightarrow C$

(d) $AB \rightarrow BA$

(e) $cAAb \rightarrow aaaa$

8. Which one of the following statements is *false*?

   (a) The union of two regular languages is regular.

   (b) The complement of a regular language is regular.

   (c) The union of two context-free languages is context-free.

   (d) The intersection of two recursively enumerable languages is recursively enumerable.

   (e) The complement of a recursively enumerable language is recursively enumerable.

9. What is Frege's principle of compositionality?

   (a) The meaning of a complete sentence must be explained in terms of its syntactic tree.

   (b) The meaning of a complete sentence must be obtained via function application.

   (c) The meaning of a complete sentence must be explained in terms of the meanings of its subsentential parts, including those of its singular terms.

   (d) The meaning of a complete sentence must be computed using lambda expressions.

   (e) The meaning of a sentence is a series of beta reduction operations.

10. How many word types and tokens are included in the following sentence:

    > I stand here today humbled by the task before us grateful for the trust you have bestowed mindful of the sacrifices borne by our ancestors

    (a) 22 types and 25 tokens

    (b) 12 types and 25 tokens

    (c) 22 types and 8 tokens

    (d) 6 types and 32 tokens

    (e) 25 types and 25 tokens

11. Which of the following statements about natural languages is the most specific *true* statement?

(a) Natural languages are a subset of unrestricted languages.

(b) Natural languages are a subset of context-free languages.

(c) Natural languages are a subset of regular languages.

(d) Natural languages are a subset of context-sensitive languages.

(e) Natural languages are a subset of both regular and unrestricted languages.

12. What is the correct meaning representation for the sentence *Every female dentist extracted a rotten tooth*?

(a) $\forall x.\text{dentist}(x) \wedge \text{female}(x) \wedge \exists y.\text{tooth}(y) \wedge \text{rotten}(y) \wedge \text{extracted(x, y)}$

(b) $\forall x.\text{dentist}(x) \wedge \text{female}(x) \Rightarrow \exists y.\text{tooth}(y) \Rightarrow \text{rotten}(y) \wedge \text{extracted(x, y)}$

(c) $\forall x.\text{dentist}(x) \wedge \text{female}(x) \Rightarrow \exists y.\text{tooth}(y) \wedge \text{rotten}(y) \wedge \text{extracted}(x, y)$

(d) $\exists x.\text{dentist}(x) \wedge \text{female}(x) \Rightarrow \forall y.\text{tooth}(y) \wedge \text{rotten}(y) \wedge \text{extracted}(x, y)$

(e) $\exists x.\text{dentist}(x) \wedge \text{female}(x) \wedge \forall y.\text{tooth}(y) \wedge \text{rotten}(y) \Rightarrow \text{extracted}(x, y)$

13. Which of the following properties holds for probabilistic context-free grammars?

(a) The sum of the probabilities of all rules in the grammar has to be one.

(b) The sum of the probabilities of all rules with the same right-hand side has to be one.

(c) The probability of the parse of a sentence is the sum of the probabilities of all the rules used to derive this parse.

(d) The probability of a sentence is the sum of the probabilities of all its parses.

(e) The probability of a sentence is the product of the probabilities of all its parses.

14. Which one of the following is **not** a garden path sentence?

(a) The prime number few.

(b) I convinced her children are noisy.

(c) The horse that raced past the barn fell.

(d) The old man the boat.

(e) We painted the wall with cracks.

15. Let w_1^n denote a sequence of $1 \ldots n$ words and t_1^n a sequence of $1 \ldots n$ parts of speech. Which of the following formulas should be used to calculate $P(t_1^n|w_1^n)$?

(a) $P(t_1^n|w_1^n) = P(t_i|t_{i-1})$

(b) $P(t_1^n|w_1^n) = \prod_{i=1}^{n} P(w_i|t_i) \prod_{i=1}^{n} P(t_i|t_{i-1})$

(c) $P(t_1^n|w_1^n) = P(t_1^n)P(w_1^n)$

(d) $P(t_1^n|w_1^n) = \prod_{i=1}^{n} P(w_i|t_i, t_{i-1})$

(e) $P(t_1^n|w_1^n) = \prod_{i=1}^{n} P(w_i, w_{i-1}|t_i)$

16. Which of the following lambda expressions has the same truth value as:
$$(\lambda P \lambda Q.\exists x.(P(x) \wedge Q(x))(\text{man}))(\lambda z.\text{yawn}(z))$$

(a) $(\lambda P \lambda Q.\exists.(P(x) \wedge Q(x)))(\text{yawn}(\text{man}))$

(b) $(\lambda P.\exists x.(\text{yawn}(x) \wedge P(x)))(\text{dog})$

(c) $(\lambda P.\exists x.(\text{yawn}(x) \wedge P(x)))(\text{man})$

(d) $\text{yawn}(\text{man})$

(e) $\exists x.(\text{man}(x) \wedge \text{yawn}(x))$

.

17. Which of the following statements about parsing algorithms is **false**?

(a) Recursive descent parsing is top-down and depth-first.

(b) LL(1) parsing can be applied to any context-free grammar in Chomsky normal form.

(c) The CYK algorithm is a bottom-up chart parsing algorithm.

(d) The Earley algorithm is a bottom-up chart parsing algorithm.

(e) The Earley algorithm uses top-down prediction to avoid building unnecessary structure.

18. Consider the following grammar for which the start symbol is S, and the lowercase English words are terminals:

S	→	NP NP | VP
NP	→	N | Det N
VP	→	V NP
V	→	book, eat
N	→	book, flight
Det	→	the

What is the complete set of analyses produced by this grammar for the sentence *book the flight*?

(a) [S [VP [V book] [NP [Det the [N flight]]]]

(b) [S [NP [N book] [NP [Det the] [N flight]]]]

(c) [S [VP [V book] [NP the [N flight]]]] and
[S [NP [N book] [NP [Det the] [N flight]]]]

(d) [S [VP [V book] [NP [Det the [N flight]]]]]] and
[S [NP [N book] [NP [Det the] [N flight]]]]

(e) [S [VP [V book] [NP [Det the [N flight]]]]]

19. Which of the following statements about Hidden Markov Models is **false**?

(a) In a Hidden Markov Model the sequence of states passed through is hidden.

(b) A Hidden Markov Model is a finite state transducer.

(c) A Hidden Markov Model has probabilities or weights on the arcs.

(d) A Hidden Markov Model is not a CYK parser.

(e) A Hidden Markov Model is defined by set of states and set of transitions between states according to input observations.

20. Which of the following English words is **not** an open-class word?

(a) drink

(b) time

(c) exam

(d) therefore

(e) can

PART B
ANSWER TWO QUESTIONS FROM PART B. Use a separate script book for each question.

1. Suppose that some system can perform sequences of actions drawn from the set $\Sigma = \{a, b, c\}$. We will be concerned here with enforcing two restrictions on such sequences: no two a-actions can occur consecutively; and between any two b-actions there must be at least one a-action. We shall provide DFAs to enforce each of these constraints separately, and then use general methods to obtain a DFA and regular expression that enforce both simultaneously.

   (a) Draw a DFA M_1 over Σ with three states that accepts precisely those strings that do not contain a consecutive subsequence aa. [*3 marks*]

   (b) Draw a DFA M_2 over Σ with three states that accepts precisely those strings in which between any two occurrences of b there is at least one occurrence of a (there may be intervening occurrences of c as well). [*3 marks*]

   (c) Suppose L_1, L_2 are the languages defined by M_1, M_2 respectively. A machine defining the language $L_1 \cap L_2$ may be obtained in the obvious way as the *product* of M_1 and M_2: states of this machine are pairs (q_1, q_2) where q_1 is a state of M_1 and M_2.

   Apply this construction to obtain a DFA with nine states that accepts $L_1 \cap L_2$. You will find it helpful to arrange the states in a 3x3 grid. To assist you in adding the correct transitions, you may also find it useful to draw a copy of M_1 on the left of the grid with a state for each row, and a copy of M_2 above the grid with a state for each column. [*9 marks*]

   (d) By applying minimization to the DFA constructed in part (c), or otherwise, draw a *minimal* DFA that accepts $L_1 \cap L_2$. [*4 marks*]

   (e) Now suppose we wish to convert the DFA from part (d) into a *regular expression* (in mathematical notation).

   Ignoring the 'garbage state' in this DFA (i.e. the one from which there are no paths to accepting states), write down an equation for each state, and solve these equations with the help of Arden's rule to obtain a regular expression corresponding to the DFA. [*6 marks*]

2. (a) Explain informally what it means for a context-free grammar to be an *LL(1) grammar*. What is the advantage of LL(1) grammars over general context-free grammars? [*3 marks*]

Now consider the following grammar, which may be used to generate descriptions of desserts on a restaurant menu. The grammar has a single non-terminal D (the start symbol), and terminals

$$\mathsf{Noun},\ \mathsf{Adjective},\ \texttt{and},\ \texttt{in},\ \texttt{a},\ \texttt{sauce}$$

Here Noun and Adjective stand for lexical classes as follows:

$$\begin{aligned} \mathsf{Noun} &= \{\texttt{banana},\ \texttt{chocolate},\ \texttt{hazelnut},\ \texttt{mango},\ \texttt{strawberry}\} \\ \mathsf{Adjective} &= \{\texttt{chopped},\ \texttt{fresh},\ \texttt{green},\ \texttt{rich},\ \texttt{stuffed}\} \end{aligned}$$

The productions of the grammar are as follows:

$$D \ \rightarrow\ \mathsf{Noun} \ \mid\ \mathsf{Adjective}\ D \ \mid\ D\ \texttt{and}\ D \ \mid\ D\ \texttt{in a}\ D\ \texttt{sauce}$$

(b) Restaurant menus are frequently ambiguous. Using the above grammar, draw all possible syntax trees for the following phrase:

`fresh strawberry and mango in a hazelnut sauce`

[*5 marks*]

(c) Write down an LL(1) grammar that defines the same language as the above grammar. Explain why, in general, LL(1) grammars are *not* particularly well suited to the description of natural languages. [*7 marks*]

(d) Calculate the First and Follow sets for each of the non-terminals in your grammar from part (c). [*8 marks*]

(e) Using the First and Follow sets, or directly by inspection, construct the LL(1) parse table for your grammar. [*7 marks*]

3. Consider the following grammar G1:

S	→	NP VP
S	→	VP
NP	→	Det NP
NP	→	PN N
VP	→	V NP
Det	→	the
N	→	run | marathon
V	→	run
PN	→	Edinburgh

(a) Describe the three types of entries that can occur in a chart generated by the Earley algorithm. [*3 marks*]

(b) Give the final chart generated by the Earley algorithm when parsing the sentence *run the Edinburgh marathon* with G1. Remember that the final chart contains all edges added during the parsing process. Represent your chart as a directed acyclic graph. [*8 marks*]

(c) Now the following rule is added to the grammar: NP $\rightarrow \epsilon$. Does this type of rule cause any problems for the Earley algorithm? Augment your chart from the previous question with the new rule to illustrate your claim. [*5 marks*]

(d) Assume you want to use the Earley algorithm to parse probabilistic context-free grammars. Give a modification of the algorithm that uses the chart to keep track of the probabilities of partial analyses. Specify how partial analysis probabilities are computed. [*6 marks*]

(e) To illustrate the algorithm you developed in the previous question, use it to parse the sentence *run the Edinburgh marathon* with G1. Assume that the rules in G1 are equiprobable, i.e., if there are n rules with the same left-hand side, then the probability of each rule is $1/n$. [*3 marks*]