UNIVERSITY OF EDINBURGH

FACULTY OF SCIENCE AND ENGINEERING

COMPUTER SCIENCE 2Ah

Monday 27 May 2002

9:30 to 12:00

Examiners

D.K. Arvind (Chair)

R. Dyckhoff (External)

INSTRUCTIONS TO CANDIDATES

Section A questions are worth 10 marks each.

Section B questions are worth 30 marks each.

Answer all FOUR questions in Section A.

Answer any TWO of the three questions in Section B.

If you are in the third or later year of your degree course, please tick the box on the front cover of the script book.

Section A

1. (a) Explain the principles of *spatial locality* and *temporal locality*. How do these principles relate to the operation of caches?

   [5 marks]

   (b) Using a diagram show the organisation of the memory hierarchy of a modern computer system with 2 levels of cache, main memory and disk. Describe the main differences between the upper and the lower levels of the hierarchy.

   [5 marks]

2. The following grammar G describes string expressions in some programming language.

Start symbol	S
Nonterminals	S E
Terminals	+ " var text

$$S \rightarrow E \qquad E \rightarrow \text{var}$$
$$S \rightarrow S + E \qquad E \rightarrow \text{" text "}$$

   (a) Write down a parse tree under this grammar for the following sentence.

   " text " + var + " text "

   [1 mark]

   (b) When is a grammar *left recursive*? Show that G is left recursive. Explain briefly why this causes problems for a top-down parser.

   [4 marks]

   (c) Construct an equivalent grammar G' that is suitable for top-down parsing without lookahead. Give the parse tree for the sentence above, under your new grammar.

   [5 marks]

3. (a) Describe the four main *phases* of the Rational Unified Process describing the purpose of each phase.

   [5 marks]

   (b) What is the role of the *workflows* in the Rational Unified Process? Illustrate your answer using two of the *core workflows*.

   [5 marks]

4. (a) Let $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$, and let $L_0 \subseteq \Sigma^*$ be the language described by the following regular expression:

$$\mathtt{ab}(\mathtt{ab})^* + \mathtt{c}(\mathtt{a} + \mathtt{b})^*$$

Design a DFA that recognises the language L_0. Draw the transition diagram of this DFA.

[5 marks]

Recall that a *prefix* of a word $\mathtt{a}_1 \dots \mathtt{a}_n$ is a word $\mathtt{a}_1 \dots \mathtt{a}_k$, where $0 \leq k \leq n$. For example, the prefixes of abac are ε, a, ab, aba, and abac. For a language $L \subseteq \Sigma^*$, let AVOID(L) be the language consisting of all words $w \in \Sigma^*$ such that no prefix of w is in L.

Let L_0 be the language introduced in part (a).

(b) Give 2 words in Σ^* that are contained in AVOID(L_0) and two words in Σ^* that are not contained in AVOID(L_0).

[2 marks]

(c) Give either a DFA or a regular expression for the language AVOID(L_0).

[3 marks]

Section B

5. (a) Briefly explain the following terms, as applied to parsers for programming languages.

   - Top down and bottom up.
   - Lookahead and backtracking
   - Predictive and table-driven.

   [6 marks]

   (b) Give a very short description of the tools lex and yacc: say what they do, and outline how to use them to build a language processing system.

   [4 marks]

   (c) The following grammar describes a fragment of the control language for a vector graphics engine.

Start symbol	P
Nonterminals	$P\ C\ R\ V$
Terminals	**draw turn repeat forever** ; num var [ ]

   $$\begin{aligned} P &\to C \; ; \; P \mid \textbf{repeat} \; R \; [\; P \;] \; P \mid \varepsilon \\ C &\to \textbf{draw} \; V \mid \textbf{turn} \; V \\ R &\to V \mid \textbf{forever} \\ V &\to \mathsf{var} \mid \mathsf{num} \end{aligned}$$

   i. Here is a sample command line for the graphics engine.

   repeat forever [**draw** num ;]

   Give a derivation of this sentence according to the grammar. List all the derivation steps in order, stating the productions used, and draw the parse tree.

   [8 marks]

   ii. Compute the *First* and *Follow* sets for this grammar. Use them to draw up a predictive parsing table for the grammar.

   [12 marks]

6. (a) Explain the difference between the virtual address space and the physical address space in a virtual memory system.

[5 marks]

(b) Discuss the main advantages of using virtual memory systems.

[10 marks]

(c) Describe the page-table mechanism for translating virtual to physical addresses.

[5 marks]

(d) Explain why the page-table is inefficient with respect to storage overhead and translation time. How do modern computer systems avoid using the page-table on some memory references and thus reduce translation time?

[10 marks]

7. In this question you are asked to contrast the Rational Unified Process with a different approach to system development. In part (7a) you are asked to describe a different system development process, in part (7b) you are asked to describe two contrasting system development scenarios and in part (7c) you are asked to use these scenarios to contrast the RUP with your chosen, contrasting, approach.

   (a) The Rational Unified Process provides a framework for describing system development processes. Provide brief notes describing a different system development process you are familiar with.

   [10 marks]

   (b) Briefly describe two contrasting system development scenarios that could be used to illustrate the strengths and weaknesses of RUP compared with the process described in your answer to part (7a) of this question.

   [10 marks]

   (c) Use your scenarios described in your answer to part (7b) of this question to provide a comparison of the strengths and weaknesses of the RUP and the development process described in your answer to part (7a) of this question. Justify your answer with reference to your scenarios.

   [10 marks]