UNIVERSITY OF EDINBURGH

FACULTY OF SCIENCE AND ENGINEERING

MASTER OF SCIENCE IN COMPUTER SCIENCE

FORMAL PROGRAMMING LANGUAGE SEMANTICS

Thursday 20 April 2000

2:00 to 3:30

Examiners:

I. Watson (External)

R.N. Ibbett (Chairman)

INSTRUCTIONS TO CANDIDATES

Answer any TWO questions.

All questions carry equal weight.

1. (a) Write out a sound and complete set of proof rules for proving partial correctness assertions of the form $\{\phi\}\, c\, \{\psi\}$, where c is a command from a simple while language with the following syntax:

$$c \quad ::= \quad X := a \;\mid\; \textbf{if } b \textbf{ then } c_1 \textbf{ else } c_2 \;\mid\; c_1 \,;\, c_2 \;\mid\; \textbf{while } b \textbf{ do } c$$

(Here a and b range over arithmetic and boolean expressions respectively. We only use standard such expressions, assuming that they have the obvious interpretations.)

[10 marks]

(b) Let P be the following program:

$$\begin{array}{l} Y := 0\,; \\ Z := 1\,; \\ \textbf{while } X \neq 0 \textbf{ do} \\ \qquad\qquad Y := Y + Z\,; \\ \qquad\qquad Z := Z \times 2\,; \\ \qquad\qquad X := X - 1 \end{array}$$

Use your proof rules to give a detailed verification of the partial correctness assertion below

$$\{X = x\}\, P\, \{Y = 2^x - 1\}$$

where x is an auxiliary variable. (Hint: You may like to use the condition $\{Z = 2^{(x-X)},\, Y = Z - 1\}$ as an invariant.)

[15 marks]

2. (a) Give a detailed definition of what it means for a set D, together with a binary relation $\sqsubseteq_D$, to form a *complete partial order (cpo)*. (Give full definitions of any auxiliary concepts you use.)

[7 marks]

(b) Suppose that $(D, \sqsubseteq_D)$ and $(E, \sqsubseteq_E)$ are cpos. State what it means for a function $f : D \to E$ to be continuous.

[3 marks]

(c) Given cpos $(D, \sqsubseteq_D)$, $(D', \sqsubseteq_{D'})$, the product cpo $(D \times D', \sqsubseteq_{D \times D'})$ is defined by:

$$\begin{array}{rcl} D \times D' & = & \{(d, d') \mid d \in D \text{ and } d' \in D'\} \\ (d_1, d_1') \sqsubseteq_{D \times D'} (d_2, d_2') & \textit{iff} & d_1 \sqsubseteq_D d_2 \text{ and } d_1' \sqsubseteq_{D'} d_2' \end{array}$$

Give a detailed verification that this is indeed a cpo.

[7 marks]

(d) Suppose given a function $f : D \times D' \to E$ satisfying:

i. for each $d \in D$ the function mapping $d' \in D'$ to $f(d, d')$ is a continuous function from $(D', \sqsubseteq_{D'})$ to $(E, \sqsubseteq_E)$; and

ii. for each $d' \in D'$ the function mapping $d \in D$ to $f(d, d')$ is a continuous function from $(D, \sqsubseteq_D)$ to $(E, \sqsubseteq_E)$;

Prove that f is a continuous function from $(D \times D', \sqsubseteq_{D \times D'})$ to $(E, \sqsubseteq_E)$.

You may use, without proof, the following fact. Suppose, for all $i, j \in \mathbb{N}$, there are elements $e_{ij} \in E$, such that $i \le i'$ and $j \le j'$ implies $e_{ij} \sqsubseteq_E e_{i'j'}$. Then $\bigsqcup_i (\bigsqcup_j e_{ij}) = \bigsqcup_i e_{ii}$.

[8 marks]

3. (a) Explain the difference between a call-by-value (eager) evaluation strategy and a call-by-name (lazy) evaluation strategy for evaluating function application in functional languages. For each of the two evaluation strategies, illustrate your answer with example rules from an operational semantics using the strategy.

[10 marks]

(b) Choosing one of the two strategies, specify an appropriate syntax for a recursion operator and give the rule, from the operational semantics, that defines the evaluation behaviour of recursion.

[5 marks]

(c) Use your syntax for recursion to write a functional program calculating the function $f : \mathit{int} \rightarrow \mathit{int}$ defined by:

$$f(x) \quad = \quad \begin{cases} 1 & \text{if } x \leq 1 \\ f(x-1) \times (f(x-2)+1) & \text{otherwise} \end{cases}$$

[3 marks]

(d) Give a detailed derivation in the operational semantics showing that your program correctly calculates $f(3) = 4$.

[7 marks]