UNIVERSITY OF EDINBURGH

FACULTY OF SCIENCE AND ENGINEERING

COMPUTER SCIENCE THREE

COMPILING TECHNIQUES

Friday 8 September 2000

14:00 to 15:30

Examiners:

Prof. K. Turner (External)

Prof. G. Brebner (Chairman)

INSTRUCTIONS TO CANDIDATES

Answer any TWO questions.

All questions carry equal weight.

1. (a) Compiler programs are often structured so that the translation from the source language to the target language is subdivided using a series of intermediate languages. What are the advantages of this approach? What are the disadvantages? [*8 marks*]

(b) Consider the following grammar. [Recall that the dollar sign denotes the end-of-file marker.]

$$
\begin{aligned}
S' &\rightarrow S\$ \\
\\
S &\rightarrow (L) \\
S &\rightarrow x \\
\\
L &\rightarrow S \\
L &\rightarrow L, S
\end{aligned}
$$

(i) Use this grammar to construct a *rightmost* derivation which begins at S' and ends with the sentence $(x, (x))$. Show the intermediate steps in your work. [*7 marks*]

(ii) Draw the LR(0) state machine for this grammar. Each state should be shown as a list of LR(0) items. [*10 marks*]

2. (a) Activations of functions within a program are allocated a *stack frame*. What is the layout of a typical stack frame on the stack? Identify the kinds of data values which are stored there. [*6 marks*]

(b) Computer manufacturers often suggest a preferred frame layout to be used on their architecture. What is the benefit to be gained by using the suggested frame layout? [*3 marks*]

(c) In a language such as Tiger or C, what are the conditions under which a local variable of a function is said to *escape* from the function? If a variable escapes then in what way does this influence the allocation of storage for that variable? [*6 marks*]

(d) The following Tiger function has formal parameters `a` and `b` and local variable `A`.

```
function f (a : int, b : int) : int =
let  type intArray = array of int
     var A := intArray[12] of 0
     function g (A : intArray) : int = A[a]
in   (A[b] := a; g (A))
end
```

(i) Which of the three variables `a`, `b` and `A` escape from the function? Explain your answer in the case of each variable. [*6 marks*]

(ii) What optimisation could be applied to the function `f` in order to reduce the number of variables which escape? Which of `a`, `b` and `A` would be affected by this optimisation? [*4 marks*]

3. (a) Modern compilers often include sophisticated algorithms to make good use of the available registers on the target machine. Briefly explain this part of the compilation process. What costs are incurred if registers are not used well? [*6 marks*]

   (b) Some machine architectures define *caller-save* and *callee-save* registers. Explain the significance of this distinction. In what circumstances might values held in these registers *not* be saved? [*8 marks*]

   (c) Explain the *dangling reference* problem. Illustrate your answer with a function expressed in the C programming language. Recall that the C statement syntax for taking the address of a variable `x` is `&x`. [*5 marks*]

   (d) A function such as `printf` in C is unusual because it can take a variable number of arguments. What restriction does this place on the allocation of space to function parameters in the C language? What constraints does it place on compiler writers implementing a compiler for C? [*6 marks*]