GX1313

OLLSCOIL NA hÉIREANN
THE NATIONAL UNIVERSITY OF IRELAND

NATIONAL UNIVERSITY OF IRELAND, GALWAY

SPRING EXAMINATIONS 1999/2000

Third University Examination in Information Technology

Programming Paradigms
CT331

Prof. D. Bell
Dr. G. Lyons
Ms. J. Griffith

Time allowed: **Two hours**

Answer any THREE questions
All questions carry equal marks

Q. 1. (i) Discuss the desirability (or otherwise) of the following programming language features:

- consistency and common notations.
- orthogonality.
- extensibility.
- portability.

(ii) Every programming language must also have associated software which translates source code to machine code. Describe the main stages involved in the translation of source code to machine code.

(iii) Discuss the advantages and disadvantages of incorporating code optimisation techniques during the translation of source code to machine code. Give examples of where code optimisation techniques can be easily incorporated.

Q. 2. (i) Describe the main data structure available in the functional programming language SCHEME.

(ii) Control in SCHEME is mainly achieved through recursion. Explain what is meant by a well-defined recursive function and discuss why tail recursive functions are used. Write non tail recursive and tail recursive versions of a function in SCHEME to calculate the sum of the first n numbers.

```
e.g. sum(3)  returns 6 (as 1 + 2 + 3 = 6)
```

(iii) Discuss the need for garbage collection routines in functional languages and outline the steps involved in any one algorithm which can be used for garbage collection.

Q. 3. (i) By the use of examples describe what is meant by:
(a) cycle-free grammars.
(b) ambiguous grammars.

(ii) Describe Chomsky's hierarchy of formal grammars by discussing:
- the associated grammars in each category, and
- the machine used to recognise the grammars in each category.

(iii) Given the following grammar:

```
<N> ={X, Y}
<T> = {x, y}
S = Y
P =
      <Y> ::= <X><X>
      <A> ::= x<X>
      <X> ::= y
```

(a) Is the grammar LL(1)? Explain your answer.
(b) Using recursive descent top-down parsing:
- parse the sentence *xxyy*.
- draw the associated parse tree.

Q. 4. (i) Define the list data structure in PROLOG and outline its representation.

(ii) Write code in PROLOG to perform the following list operations, explaining the steps taken for each function:

- Delete an item from a list.
- Merge two lists.

(iii) Given the following PROLOG program which contains rules that define a function `range` such that given the goal `range(X, Y)` the following is returned:

Y = 0 if X = 3.
Y = 2 if X is >= 3 and X < 6.
Y = 4 if X >= 6.

```
range(X, 0) :- X < 3.
range(X, 2) :- 3 =< X, X < 6.
range(X, 4) :- 6 =< X.
```

(a) Discuss the inefficiencies in the program.
(b) Outline the changes which can be made to the rules to lead to a more efficient program.
(c) Discuss whether the changes you made to the program in part (b) affect the procedural and declarative meaning of the program.

Q. 5. (i) Discuss a representation of binary trees in SCHEME.

(ii) Explain and outline the main steps involved in searching for an item in a binary tree using:

- depth first search, and
- breadth first search.

(iii) Write code in SCHEME to implement either depth first search *or* breadth first search using the binary tree representation developed in part (i) and the algorithm developed in part (ii).