

# 2004 Paper 4 Question 1

## Compiler Construction

- (a) A context-free grammar can be formally defined as a 4-tuple. Give a precise statement of what the components are. [2 marks]
- (b) Explain the difference between a grammar and the language it generates. [2 marks]
- (c) Explain what makes a grammar ambiguous, with reference to the grammar which may be commonly expressed as a “rule”

$$E ::= 1 \mid 2 \mid X \mid E + E \mid E * E \mid -E$$

where  $X$  is an identifier. [2 marks]

- (d) For the “rule” in part (c), give a formal grammar containing this “rule” and adhering to your definition in part (a). [2 marks]
- (e) Give non-ambiguous grammars each generating the same language as your grammar in part (d) for the cases:
- (i) “-” is most tightly binding and “+” and “\*” have equal binding power and associate to the left.
- (ii) “-” is most tightly binding and “+” and “\*” have equal binding power and associate to the right.
- (iii) “-” binds more tightly than “+”, but less tightly than “\*”, with “+” left-associative and “\*” right-associative so that “ $-a + -b * c * d + d$ ” is interpreted as “ $((-a) + (-(b * (c * d)))) + d$ ”.

[2 marks each]

- (f) Give a simple recursive descent parser for your grammar in part (e)(iii) above which yields a value of type `ParseTree`. You may assume operations *mkplus*, *mktimes*, *mkneg* acting on type `ParseTree`. [6 marks]