419433

UNIVERSITY OF YORK

BA and BSc Degree Examinations 2000-2001

DEPARTMENT OF LANGUAGE AND LINGUISTIC SCIENCE

L433: Introduction to Computational Linguistics

Time allowed: $1\frac{1}{2}$ hours

Answer ALL questions

Total marks: 80

(1) Explain in English what is meant by the following expressions.

(a) $P \subset N \times (N \cup T)^*$ (5 marks)

(b) $A \Rightarrow aB\alpha$ (5 marks)

(2) Given the following grammar,

$\mathsf{S} \rightarrow \mathsf{NP\ VP}$	*Fido*: NP
$\mathsf{NP} \rightarrow \mathsf{D\ N}$	*the*: D
$\mathsf{VP} \rightarrow \mathsf{V\ NP}$	*man*: N
$\mathsf{PP} \rightarrow \mathsf{P\ NP}$	*in*: P
$\mathsf{NP} \rightarrow \mathsf{NP\ PP}$	*park*: N
	saw: V

(a) Provide a bottom-up, left-to-right derivation for the string *Fido saw the man in the park*. (9 marks)

(b) State what kind of derivation it is. (1 mark)

(c) What problem would a top-down parser encounter in parsing the string *Fido saw the man in the park* as defined by the above grammar, and why? (10 marks)

(3) Under what circumstances would two grammars G_1 and G_2 be said to be *weakly equivalent*? (5 marks)

(4) (a) What kind of language is $a^n b^n$? (1 mark)

(b) How is this class of languages defined? (3 marks)

(c) What class of languages does the language defined by the following grammar belong to? (5 marks)

$$\begin{aligned} \mathsf{S} &\rightarrow \mathsf{a}\ \mathsf{S} \\ \mathsf{S} &\rightarrow \mathsf{a} \\ \mathsf{S} &\rightarrow \mathsf{b}\ \mathsf{S} \\ \mathsf{S} &\rightarrow \mathsf{b} \end{aligned}$$

(d) What is the relationship between the classes of languages referred to in questions (a) and (c) above? (2 marks)

(5) (a) What is the difference between a recogniser and a parser? (4 marks)

(b) What is the significance of this difference for the time complexity results for context-free languages? (4 marks)

(6) Outline the operation of a shift-reduce parser and show the states of the stack and buffer during a parse of the string *Fido saw the man* using the grammar in question (2) above. (12 marks)

(7) Immediately below is the Prolog code for a top-down, depth-first left-to-right recogniser. Compile the parser over the grammar in question (2) above. (10 marks)

```
td_parse(Cat,[Word|RestofString],RestofString) :-
   (Cat===> [Word]).

td_parse(Mother, S0,S) :-
    (Mother ---> Daughters),
   td_parse_dtrs(Daughters,  S0, S).

td_parse_dtrs([],S, S).

td_parse_dtrs([Cat|Cats], S0, S) :-
   td_parse(Cat, S0, S1),
   td_parse_dtrs(Cats, S1, S).
```

(8) (a) How does a Definite Clause Grammar (DCG) differ from a CF-PSG? (2 marks)

(b) What is the weak generative capacity of DCGs? (2 marks)

(c) Write a minimal DCG in either BH or Prolog Grammar Rule Notation (making clear which you are using) that will generate the following sentences of English. (10 marks)

(i) These books are cheap

(ii) This book is cheap

(iii) The book is cheap

(iv) The books are cheap

(v) A book is cheap

(vi) The sheep are cheap

(vii) The sheep is cheap