

## NOTES

To supplement past examination papers, I have collected together here old examination questions on NLP from when I taught this at Aston University. There are some differences to be noted:

- Prolog was then included; ignore!
- Marks are based on 20 marks per question, answer 3 from 5 in 2 hours, i.e. 20 marks = 33%, 60 marks = 100%.
- What may appear to be ‘new’ problem questions may have been part of lab. classes, so would have been less difficult than they might appear.

## General

### 1997/UG/6

6. Two people are alone in a room. One says to the other *Have I asked you your income?*

Three ‘levels’ at which such a sentence can be processed are:

- morphological
- semantic
- pragmatic.

For EACH of these levels, explain what is involved, using the sentence given above in at least one of your examples, and comment briefly on the feasibility of computer-based processing.

(20 marks)

### 1999/UG/1

1. “The keyboard will soon be as obsolete as the card punch and users will communicate with computers the way they do with each other – by means of speech.”  
Discuss this proposition.

(20 marks)

### 2000/UG/1

1. “Natural Language Processing by computer is currently neither possible nor impossible but somewhere between.”

Discuss this proposition, paying careful attention to the evidence you use. State and justify your conclusion.

(20 marks)

### 2000/UG/3

3. a) In lectures, three broad ‘levels’ of natural language processing were distinguished: sound systems, grammar and meaning. Two people are alone in a room. One says to the other *Could you tell me your income?* Using the italicized sentence as an example, briefly explain the processing required at each level if a robot were to be able to ‘understand’ spoken natural language.

(8 marks)

## Sound Systems

1997/MSc/3

3. a) English speakers have no difficulty in distinguishing the two sequences of sounds described in the IPA as [rʌm] and [rʌn], but normally consider [ɪmpʊt] and [ɪnpʊt] to be the same word.

Explain carefully the meaning of the terms ‘phone’ and ‘phoneme.’ Give TWO examples in your explanation; the above information should be used as one of them.

(10 marks)

- b) In fast speech in some dialects of English, the pronunciation of the last syllable of words like *teacher* or *weather* (usually represented in writing by *er*) depends on the first phone of the following word. Some examples:

... <i>teacher of</i> ...	[..ər ɒ..]
... <i>weather and</i> ...	[..ər æ..]
... <i>weather outside</i> ...	[..ər aʊ..]
... <i>teacher was</i> ...	[..ə w..]
... <i>teacher may</i> ...	[..ə m..]
... <i>weather can</i> ...	[..ə k..]

- i) Suggest, in your own words, phonological rule(s) to explain the differences in pronunciation.

(2 marks)

- ii) Assuming the existence of `p_process/2`, defined below, and the `phone/5` definitions on Page 3, write `p_rule/2` which implements in Prolog the rule(s) you set out in (i) above.

```
p_process([], []).
p_process(InList, OutList) :-
    p_rule(InList, TempList), InList \= TempList,
    p_process(TempList, OutList).
p_process([Head|InTail], [Head|OutTail]) :-
    p_process(InTail, OutTail).
```

(2 marks)

- c) What are the consequences of phone/phoneme differences for speech recognition?

(6 marks)

**1998/MSc/3**

3. a) With the aid of appropriate examples, explain carefully the meaning of the terms 'phone' and 'phoneme'.

(6 marks)

- b) The table below is intended for use in a TTS (text-to-speech) system. It gives a possible phonemic representation for some English words, together with their approximate pronunciation in 'Standard English English' (SEE) and 'Standard American English' (SAE).

	Phonemic <u>Representation</u>	<u>SEE</u>	<u>SAE</u>
<i>tort</i>	/tɔrt/	[tɔt]	[tɔrt]
<i>terror</i>	/tɛrər/	[tɛrə]	[tɛrər]
<i>rid</i>	/rɪd/	[rɪd]	[rɪd]
<i>rat</i>	/ræt/	[ræt]	[ræt]
<i>writer</i>	/raɪtər/	[raɪtə]	[raɪdər]
<i>rider</i>	/raɪdər/	[raɪdə]	[raɪdər]
<i>doer</i>	/duər/	[duə]	[duər]

- i) Suggest two GENERAL spelling (grapheme) → phoneme rules which are illustrated in the table and which could be incorporated into a TTS system.

(2 marks)

- ii) On the evidence of the table, the phonemes /r/ and /t/ are subject to different phonological rules in SEE and SAE. State these rules. (You may use any appropriate notation, including Prolog code.) On the basis of your rules how should the TTS system pronounce the word *latterly* (phonemic representation /lætərli/) in SEE and SAE?

(6 marks)

- c) TTS and STT (speech-to-text) systems can use either the method implied in the table in (b), i.e. spelling ↔ phonemes ↔ phones, or the more direct spelling ↔ phones approach. Briefly outline the advantages and disadvantages of using phonemes as an intermediate in

- i) TTS systems  
ii) STT systems.

(6 marks)

**1998/UG/4**

4. a) The MacinTalk text-to-speech (TTS) system has two components: one which converts text to 'phoneme codes' and another which produces phones from the phoneme codes.

i) Explain the meaning of the term 'phoneme' in this context. Illustrate your answer by discussing TWO appropriate examples. (6 marks)

ii) What are the advantages of using phoneme codes as an intermediate stage in TTS, as compared to direct conversion of text to sound? (4 marks)

b) A TTS system is designed to speak the plural of English nouns in the following way. The phonemic representation of the singular form of the noun is obtained via some other process. The phoneme /z/ is added to this phonemic representation and then phonological rules are used to generate the correct pronunciation.

The table below shows the pronunciation which should be generated for some English nouns.

<u>Noun</u>	<u>Phonemic Representation of Singular</u>	<u>Phonemic Representation of Plural</u>	<u>Pronunciation of Plural</u>
<i>cuff</i>	/kʌf/	/kʌfz/	[kʌfs]
<i>dove</i>	/dʌv/	/dʌvz/	[dʌvz]
<i>death</i>	/deθ/	/deθz/	[deθs]
<i>lathe</i>	/leɪð/	/leɪðz/	[leɪðz]
<i>bus</i>	/bʌs/	/bʌsz/	[bʌsɪz]
<i>nose</i>	/noʊz/	/noʊzz/	[noʊzɪz]
<i>bush</i>	/bʊʃ/	/bʊʃz/	[bʊʃɪz]

State in words and in 'feature set notation' the phonological rule(s) needed to generate the correct pronunciation of the plurals above. (6 marks)

c) The word *garage* is pronounced in English in at least two ways: [gæɾɑʒ] and [gæɾɪdʒ]. What problems could these variant pronunciations of *garage* cause a speech-to-text (STT) system? How might they be overcome? (4 marks)

**1999/UG/2**

2. a) The three sequences of sounds described in the IPA as [rʌm], [rʌn] and [rʌŋ] are perceived by English speakers as three distinct words (*rum*, *run* and *rung*). On the other hand, English speakers normally consider [ɪnpʊt] and [ɪmpʊt] to be the same word (*input*), and [ɪnkʌm] and [ɪŋkʌm] to be the same word (*income*). Using this information as an example, explain carefully the meaning of the terms ‘phone’, ‘phoneme’, ‘phonological rule’ and ‘assimilation’.

(10 marks)

b) A company wants a TTS system which can generate speech in different regional dialects (in order to appear more ‘customer friendly’). One proposed implementation involves creating a single dictionary giving a phonemic representation of each word and then using different phonological rules to generate the correct pronunciations in the regional dialects. The table below shows the proposed phonemic representations for some words and the corresponding pronunciations in a particular English dialect.

Word	Phonemic Representation	Pronunciation in Dialect A
<i>tart</i>	/tɑrt/	[tɑt]
<i>terror</i>	/tɛrər/	[tɛrə]
<i>rat</i>	/ræt/	[ræt]
<i>term</i>	/tɜrm/	[təm]
<i>hirer</i>	/haɪrər/	[haɪrə]
<i>hiring</i>	/haɪrɪŋ/	[haɪrɪŋ]
<i>fireman</i>	/faɪrmən/	[faimən]

i) Linguists state that in Dialect A, the following phonological rules operate:  
 anything .. vowel : /r/ → [r]  
 anything .. non-vowel : /r/ → nothing

In this context, ‘non-vowel’ includes ‘end-of-word’. Explain carefully how a program can use these rules to generate the correct pronunciations of appropriate words in Dialect A from their phonemic representations in the table. (Code is NOT required.)

(4 marks)

ii) Assuming the existence of `p_process/2`, defined below, and the `phone/5` definitions on Page 5, write one or more `p_rule/2` clauses which implement in Prolog the rules set out in (i) above. Assume `vowel/1` exists.

```
p_process([], []).
p_process(InList, OutList) :-
    p_rule(InList, TempList), InList \= TempList,
    p_process(TempList, OutList).
p_process([Head|InTail], [Head|OutTail]) :-
    p_process(InTail, OutTail).
```

(2 marks)

iii) In Dialect B, all the words in the table are pronounced exactly as their phonemic representation suggests (e.g. *fireman* is pronounced [faɪrmən]). In the same dialect, the word *iron* is pronounced [aɪrən]. However, in Dialect A, *iron* is pronounced [aɪən]. What problems will this cause with the proposed TTS system? How might they be overcome?

(4 marks)

**2000/UG/2**

2. a) TTS (text-to-speech) systems often employ a two-stage process in which phoneme codes are used as an intermediate between graphemes and allophones.

Explain carefully the meaning of the terms ‘grapheme’, ‘phoneme’ and ‘allophone’ in this context. Illustrate your explanation by considering how such a system might correctly pronounce the words *Philip* (phonetically [fɪlɪp]) and *pill* (phonetically [p<sup>h</sup>ɪl]).

(9 marks)

- b) Phonemically, a regular English verb can be considered to have its ‘third singular’ formed by the addition of /z/ and its ‘past tense’ by the addition of /d/. The actual pronunciation may be different. The table below shows some examples.

Base Verb		Third Singular		Past Tense	
Written Form	Phonemes	Phonemes	Phones	Phonemes	Phones
<i>tap</i>	/tæp/	/tæpz/	[tæps]	/tæpd/	[tæpt]
<i>rub</i>	/rʌb/	/rʌbz/	[rʌbz]	/rʌbd/	[rʌbd]
<i>hack</i>	/hæk/	/hækz/	[hæks]	/hækd/	[hækt]
<i>hug</i>	/hʌg/	/hʌgz/	[hʌgz]	/hʌgd/	[hʌgd]

State in words and in ‘feature set notation’ the phonological rule(s) needed to generate the correct pronunciation of the third singulars and past tenses above. Briefly explain, without writing code, how this process can be computerized.

(7 marks)

- c) Most current speech recognition systems do not make explicit use of phonemes or phonological rules. Instead they rely on matching sounds to stored word patterns coupled with extensive statistical prediction of word likelihood. Briefly explain ONE advantage and ONE disadvantage of omitting intermediate phonemic representations.

(4 marks)

## Morphology

### 2000/UG/3 (part)

3. b) Consider these sentences:

*I **require** more money.*  
*I keep **requiring** more money.*  
*Money is **required**.*  
*My main **requirement** is money.*  
*My **requirements** are few.*

Using the emboldened words in the above sentences as examples, briefly explain what is meant by the terms ‘lexeme’, ‘morpheme’, ‘inflectional morphology’ and ‘derivational morphology’.

(4 marks)

c) The process of affixing *s* to the written form of an English noun to form the plural can be coded in Prolog as follows.

```
% If the word has only two letters, invoke a rule.
affix_s([L1,L2],Result):-
    affix_s_rule([L1,L2],Result).
% Otherwise reduce the word in length and try again.
affix_s([Head|InTail],[Head|OutTail]):-
    affix_s(InTail,OutTail).
```

Three of the rules involved are illustrated by the words *catch*, *country*, *play* and *cat*, which become *catches*, *countries*, *plays* and *cats* respectively. Write `affix_s_rule/2` which implements these rules (and only these rules). You may need to write auxiliary predicates.

(4 marks)

d) To handle irregular forms (e.g. the plural of *man* is *men* not *mans*) involves the use of a dictionary at some stage. Explain carefully the LOGIC underlying a Prolog predicate which relates the singular and plural forms of an English noun and its consequences for morphological analysis. You do NOT need to write any code.

(4 marks)

## Syntax

1997/UG/5

5. The following fragment of a grammar describes some simple sentences in English. (Note that `nom` and `acc` are constants, `C` is a variable.)

`S` → `NP(nom) VP`

`VP` → `verb NP(acc)`

`NP(C)` → `pronoun(C)`

`NP(C)` → `noun`

`pronoun(nom)` → `you`

`pronoun(nom)` → `they`

`pronoun(acc)` → `you`

`pronoun(acc)` → `them`

`noun` → `cats`

`verb` → `like`

- a) For each sentence given below, state whether it is valid according to the above grammar. Use your answers to explain IN DETAIL how the grammar accepts or rejects such sentences.

i) *cats like them*

iii) *they like you*

ii) *cats they like*

iv) *cats like they.*

(5 marks)

- b) Write a Prolog version of the grammar above. You may either write it in full or use the `-->` operator. Your answer should assume that a lexicon has been defined which includes the entries:

`lex(they,pronoun,nom).`

`lex(they,pronoun,acc).`

`lex(you,pronoun,_).`

`lex(cats,noun).`

`lex(like,verb).`

(5 marks)

- c) Explain briefly how the code you wrote in (b) above can be extended to include a Prolog representation of the syntax tree. Give at least ONE example of a re-written clause. You do NOT need to re-write all of the code.

(3 marks)

- d) One possible method for machine translation seems to be to convert an input sentence into a syntax tree ‘decorated’ with the base words in the input language. The words could then be translated 1:1 into the output language, after which the resulting syntax tree could be used to generate an output sentence. Why doesn’t this work? What are the consequences for machine translation? Some information which may be useful in your answer is given below.

- *They like cats* would have the same word order when translated into French, but French has two words for *they*: *ils* meaning men (or a mixture of men and women) and *elles* meaning two or more women.
- *They like cats* would have to be translated into Spanish as the equivalent of *Cats please them.*

(7 marks)

**1998/MSc/4 (part)**

4. The following fragment of a grammar, written using the Prolog ‘-->’ notation, describes some ‘verb phrases’ in English.

```
vp --> verb(Comps), verbComps(Comps).  
verbComps(np) --> np.  
verbComps(np_pp) --> np, pp.  
np --> det, noun.  
pp --> prep, np.  
det --> [the].  
noun --> [book] ; [boy] ; [girl].  
prep --> [to].  
verb(np) --> [saw].  
verb(np_pp) --> [gave].
```

- a) In the sentences below, the words in square brackets make up a verb phrase. For each sentence state whether the verb phrase is valid according to the grammar given above. Use your answers to explain how the grammar enforces agreement between a verb and its complements. You are NOT expected to explain in detail how the Prolog code actually works.

- i) *The boy [saw the girl].*  
ii) *The boy [saw].*

- iii) *The girl [gave the book to the boy].*  
iv) *The girl [gave the boy the book].*

(5 marks)

**1998/UG/5**

5. The following grammar describes some simple sentences in English. Arguments are written using the Prolog conventions for variables and constants.

$S \rightarrow NP(A, \text{nom}) VP(A)$

$VP(A) \rightarrow \text{verb}(A, \text{Comps}) \text{VerbComps}(\text{Comps})$

$\text{VerbComps}(\text{np}) \rightarrow NP(\_, \text{acc})$

$NP(A, \text{Case}) \rightarrow \text{pronoun}(A, \text{Case})$

$NP(A, \_) \rightarrow \text{noun}(A)$

$\text{pronoun}(n, \text{nom}) \rightarrow I$

$\text{pronoun}(n, \text{acc}) \rightarrow me$

$\text{pronoun}(t, \text{nom}) \rightarrow she$

$\text{pronoun}(t, \text{acc}) \rightarrow her$

$\text{noun}(t) \rightarrow cat$

$\text{noun}(n) \rightarrow cats$

$\text{verb}(n, \text{np}) \rightarrow like$

$\text{verb}(t, \text{np}) \rightarrow likes$

- a) For each sentence given below, state whether it is valid according to the above grammar. Use your answers to explain IN DETAIL how the grammar accepts or rejects such sentences.
- i) *cat likes me*
  - ii) *cat likes*
  - iii) *she likes I.* (6 marks)
- b) Explain how the grammar above could be converted to Prolog code using the `-->` notation. Assume that words and their properties are to be stored in a lexicon rather than in the grammar. You need not write out all the code but should give appropriate examples. (5 marks)
- c) Extend the grammar above to handle sentences like *I say that she likes me* or *she says that I say that I like her*. You may use either the original notation or Prolog code. (4 marks)
- d) Explain how the Prolog code you wrote in (b) could be extended to generate a 'parse tree'. You need not re-write the whole code, but should give examples to illustrate your explanation. (5 marks)

**1999/UG/4**

4. The following fragment of a grammar describes some ‘verb phrases’ in English:

VP  $\rightarrow$  verb(CompType) VerbComp(CompType)

VerbComp(np)  $\rightarrow$  NP

VerbComp(np\_np)  $\rightarrow$  NP NP

NP  $\rightarrow$  noun | det noun

det  $\rightarrow$  *the*

noun  $\rightarrow$  *people*

noun  $\rightarrow$  *mice*

noun  $\rightarrow$  *cheese*

verb(np)  $\rightarrow$  *killed*

verb(np\_np)  $\rightarrow$  *gave*

- a) For each of the three verbs, give an example of a verb phrase which is valid according to the grammar and another example which has invalid complements. Using your examples, explain in some detail how the above grammar ensures that the complements of a verb are syntactically correct. (5 marks)
- b) Write a Prolog version of the grammar above. You may either write it in full or use the  $-->$  operator. Include an appropriately defined lexicon containing the six words in the grammar. (5 marks)
- c) One possible method for machine translation is to convert an input sentence into a syntax tree ‘decorated’ with the base words in the input language. The words could then be translated one-to-one into the output language, after which the resulting syntax tree could be used to generate an output sentence. Why doesn’t this work? What are the consequences for machine translation? (10 marks)

**2000/UG/4**

4. The following fragment of a grammar describes some ‘verb phrases’ in English ( $\emptyset$  means ‘nothing’, Prolog notation is used for variables):

VP  $\rightarrow$  verb(CompType) VerbComp(CompType)

VerbComp( $\emptyset$ )  $\rightarrow$   $\emptyset$

VerbComp(np)  $\rightarrow$  NP

VerbComp(np\_pp)  $\rightarrow$  NP PP

VerbComp(np\_np)  $\rightarrow$  NP NP

NP  $\rightarrow$  noun

PP  $\rightarrow$  prep NP

noun  $\rightarrow$  *money*

noun  $\rightarrow$  *John*

prep  $\rightarrow$  *to*

verb( $\emptyset$ )  $\rightarrow$  *died*

verb( $\emptyset$ )  $\rightarrow$  *killed*

verb(np)  $\rightarrow$  *killed*

verb(np\_pp)  $\rightarrow$  *gave*

verb(np\_np)  $\rightarrow$  *gave*

- a) Explain with examples how this grammar enforces the correct complements for the verbs *died*, *killed* and *gave*.  
(6 marks)
- b) Briefly explain how this grammar can be converted into Prolog code, both WITHOUT and WITH the use of the ‘-->’ convention. You do NOT need to rewrite all of the grammar but should include sufficient to show the principles involved.
- c) Consider the following examples of natural language.
- i) Tim, a child aged 3, says *I died my teddy*.
  - ii) Paul says *I never give to charity*.

Neither of the two italicised sentences appears to be consistent with the grammar above. Discuss each sentence in turn, explaining the nature of the problem and, if possible, suggesting how the syntax and then the semantics of each could be processed.

(7 marks)

## Semantics

### 1998/MSc/4 (part)

4. b) What is meant by the term ‘theta role’ ( $\theta$ -role)? Illustrate your answer by discussing the theta roles involved in the Active and Passive sentences below. Show how both syntax and theta roles contribute to the acceptability or otherwise of the sentences.

- |  |  |
|--|--|
| i) <i>The girl gave the book to the boy.</i> | iv) <i>The boy was given the book.</i> |
| ii) <i>The girl gave the boy the book.</i>   | v) <i>*The book was given.</i>         |
| iii) <i>The book was given to the boy.</i>   | vi) <i>*The girl gave to the boy.</i>  |

(7 marks)

- c) One possible approach to machine translation involves the creation of intermediate analyses involving theta roles. Outline the advantages of this method compared to one in which the intermediate is a syntactic analysis (e.g. a parse tree).

An example which may be helpful in your answer is the English sentence *I like Greece* which translates to the Greek *Mou aresi i Ellada* (literally ‘to-me it-pleases the Greece’ = ‘Greece pleases me’).

(4 marks)

- d) ‘Ellipsis and ‘anaphora’ are two further issues which need to be considered in the semantic processing of natural language. Briefly explain each, using the following sentences as examples:

- |   |
|---|
| i) <i>The girl gave the book to the boy and the pencil to the teacher.</i>      |
| ii) <i>The girl gave the pencil to the teacher but she gave it back to her.</i> |

(4 marks)

### 1998/UG/6

6. a) Consider the following two sentences (whose syntax is identical):

*This man’s shirt is quite nice, even though it’s green.*  
*This apple’s taste is quite nice, even though it’s green.*

How could a program recognize that in the first sentence *green* refers to *shirt* rather than *man*, whereas in the second sentence *green* refers to *apple* rather than *taste*? Would the method you describe enable *it* to be identified as *shirt* in the first sentence and as *apple* in the second?

(7 marks)

- b) What is meant by the term ‘theta role’ ( $\theta$ -role)? How can  $\theta$ -roles be used in a program to decide that sentences (i) and (ii) below are acceptable, whereas (iii) is not?

- |  |
|--|
| i) <i>Jane saw the eclipse.</i>          |
| ii) <i>The eclipse was seen by Jane.</i> |
| iii) <i>*The eclipse saw Jane.</i>       |

(8 marks)

- c) Suppose that the methods you have described in (a) and (b) above have been incorporated into a ‘semantic processing program’. How successful will such a program be at processing the meaning of a random sample of English text? Briefly justify your answer.

(5 marks)

**1999/UG/5**

5. a) The table below shows some possible semantic feature sets for three nouns.

<i>Jane</i>	[−COMMON, −COUNT, +ANIMATE, +HUMAN, −MALE, +FEMALE, −ABSTRACT]
<i>boy</i>	[+COMMON, +COUNT, +ANIMATE, +HUMAN, +MALE, −FEMALE, −ABSTRACT]
<i>idea</i>	[+COMMON, +COUNT, −ANIMATE, −HUMAN, −MALE, −FEMALE, +ABSTRACT]

- i) By appropriate use of rules, it is possible to reduce the number of features which need to be stored in a lexicon. Suggest THREE rules that could be used here. (3 marks)
- ii) Consider the following pairs of sentences.

*Jane told the boy about her ideas. She trusted him.*  
*Jane told the boy about her ideas. They laughed.*  
*Jane told the boy about her ideas. He liked them.*

Full semantic processing requires the pronouns *she*, *him*, *they*, *he* or *them* in the second sentence of each pair to be identified with the appropriate nouns in the first sentence of each pair. Discuss how far this can be achieved by the use of the semantic features given above. (7 marks)

- b) Explain what is meant by the term ‘theta role’ (θ-role). (2 marks)

c) A proposed lexicon entry for the verb *give*, employing θ-roles, is of the form:

*give* : complements = NP/Agent: [+ANIMATE] + NP/Patient + PP(*to*)/Recipient |  
 NP/Agent: [+ANIMATE] + NP/Recipient + NP/Patient

In the lecture course, the ‘passivization rule’ was given as:

If and only if valid active verb complements have:

- NP/Agent as the first component
- NP/<Role> as the second component

the corresponding valid passive verb complements have:

- NP/<Role> as the first component
- the remaining components of the active verb complements
- optionally PP(*by*)/Agent as the final component.

Using the proposed lexicon entry and this rule, explain how a program can generate a semantic analysis of the sentence:

*Tom was given the tie.*

(You are NOT expected to write any code.) (4 marks)

- d) Only the first of the following sentences is acceptable when spoken in a ‘normal’ intonation:

*For John’s birthday, I gave him a tie.*  
 \**For John’s birthday, I gave a tie to him.*

Why? Can such distinctions be handled by a computer program? (4 marks)

**2000/UG/5**

5. a) Briefly explain what is meant by the term 'semantic features'. (2 marks)

b) Consider the grammatical subject NP in these sentences:

<i>Everest is a mountain.</i>	<i>*The Everest is a mountain.</i>
<i>*Thames is a river.</i>	<i>The Thames is a river.</i>
<i>*Atlantic is an ocean.</i>	<i>The Atlantic is an ocean.</i>
<i>Windermere is a lake.</i>	<i>*The Windermere is a lake.</i>
<i>London is a city.</i>	<i>*The London is a city.</i>
<i>England is a country.</i>	<i>*The England is a country.</i>

Suggest semantic features for the nouns *Everest*, *Thames*, *Atlantic*, *Windermere*, *London* and *England* which can be used to account for the distribution of the determiner *the*.

(4 marks)

c) Explain what is meant by the term 'theta role' ( $\theta$ -role), distinguishing between participant and circumstantial theta roles. (4 marks)

d) A proposed lexicon entry for the verb *show*, employing  $\theta$ -roles, is of the form:

*show* : complements = NP/Agent: [+ANIMATE] + NP/Patient + PP(*to*)/Recipient |  
NP/Agent: [+ANIMATE] + NP/Recipient + NP/Patient

Explain carefully what this entry means, showing how it may be used to analyse TWO example sentences (you may ignore passive sentences).

(5 marks)

e) Only the first of the following sentences is acceptable when spoken in a 'normal' intonation (e.g. without extra stress).

*I showed him the outside* [– not the inside].  
*\*I showed the outside to him* [– not the inside].

One explanation for this is that the non-bracketed part of each sentence has different presuppositions embedded in the different verb complements. Explain. Can this be handled by a computer program?

(5 marks)