The University of Melbourne

School of Engineering

800-204 Language and Computation

Mid-Semester Test, 9 September 2010

Duration: 45 minutes

Length: This paper has 7 pages including this cover page

Instructions to students: This paper counts for 10% of your final grade. Answer all questions in the spaces provided. Use page 7 if extra space is required for any question.

Examiners: Steven Bird, Lesley Stirling

Student id:

Examiner's use only:	Part A:	Part B:	Part C:	Total:

Part A: Key Concepts

1. Give short definitions of the following terms as they are used in the context of language and computation, with the help of examples:

    (a) n-gram part-of-speech tagger

    (b) recursive syntactic structure

    (c) structural ambiguity

2. Explain the following concepts and the distinctions between them, with the help of examples:

    (a) transitive *vs* intransitive verb

    (b) syntactic well-formedness *vs* semantic interpretability

    (c) parser *vs* grammar

Part B: Language Analysis

3. Draw a tree diagram to represent the constituent structure of the sentence: *I gave a present to an old friend for her birthday*

4. What is the morphological structure of the word: *disappearance*

5. Explain the process of training a unigram tagger; show the internal data structure that would be built up while processing the following training data:

```
They/PR refuse/V to/TO permit/V us/PR to/TO obtain/V the/D refuse/N permit/N
```

Part C: Python Programming

6. Explain the purpose of the following code:

```
def mystery(s):
    d = {'v':0, 'c':0}
    for w in s:
        if w.isalpha():
            if w in "aeiou":
                key = "v"
            else:
                key = "c"
            d[key] += 1
    return float(d["v"]) / d["c"]
```

7. Suppose that a text is represented as a list of words and punctuation symbols (i.e. “tokens”), where each token is represented as a string. Assume the text `text1` is already defined. Write code to find all words of `text1` that contain at least two capital letters (e.g. *CNet, R.E.M., R&D, DayGlo*).

8. Define a function `longest(text)` that takes a text (represented as a list of tokens), and returns the length of the longest word in `text`.

Extra Work