CS3470 Assignment 2 2018/19

This assignment should be submitted in PDF format via Moodle by **2pm** on **Friday 9th November 2018**.
All the work you submit must be solely your own work.

1. [**13 marks**] Give a leftmost derivation and a rightmost derivation for the sentence `bdxdb` in the following grammar, with start symbol S. Prove that the grammar is ambiguous.

$$\begin{array}{lcl} S & ::= & A\ S\ A\ b \mid b\ d \mid S\ B \mid \epsilon \\ T & ::= & d \mid T\ b \end{array} \qquad \begin{array}{lcl} B & ::= & T\ x \mid x \mid S \\ A & ::= & \epsilon \mid A\ d \end{array}$$

2. [**20 marks**] Use the standard algorithm given in class to remove the left recursion from each the following two grammars:

```
(i)   S ::= S a |  T b  |  a       T ::= T b | S T

(ii)  S ::= B S a | S D | b            A ::= a | c D | A a
      B ::= B B   | A | S              D ::= S B | a
```

3. [**20 marks**] Calculate the FIRST sets for the non-terminals in grammar (iii).

```
(iii)  S ::= S a M | S M d | E E     M ::=  b E | S y     E ::= ε
```

Calculate the FOLLOW sets for all the non-terminals in grammars (iv) and (v).

```
(iv) S ::= S a | V b      V ::= c | b | ε
(v) S ::=  Y X        Y ::= b b S Y d | X c a       X ::= a | ε
```

4. [**28 marks**] *[CS3470 Exam 2013]* Compute the first sets of the nonterminals in the grammar Γ_3 which has nonterminals S, A, B, start symbol S and grammar rules

$$S ::= A\ d \mid v\ d \mid w\ S \qquad A ::= v\ A \mid B\ B \qquad B ::= u\ S\ A\ v \mid \epsilon$$

(Using the method given in lectures) write out, in pseudo code, recursive descent parse functions for each of the nonterminals in Γ_3. (You may assume the existence of a function `error()` which terminates the parser and outputs an error message and a function `gnt()` which reads and returns the next input symbol.)

Write out the derivation tree which corresponds to a parse of the string wvd using your parse functions.

5. [**19 marks**] *[CS3470 2014]* The following partially annontated grammar, written in **rdp**'s IBNF, defines a language fragment which generates subtractions and products of integers.

```
S:int ::= A(0 0):result [* printf("value = %i\n", result); *] .
A ::=  B [ '-' A | '+' A ] .
B ::=  C { '*' C } .
C ::=  INTEGER | '(' A ')' .
```

Add **rdp**-style attributes and semantic actions to the grammar rules for A, B and C so that the expressions are evaluated correctly, for example so that for input `5-7*(2-3)+1` the value `13` is printed. Run your grammar through **rdp** and show the output when the parser is run on the string `5-7*(2-3)+1`

..../PTO

Assessment Details And Criteria

This assignment is worth 5% of the final course mark. The questions on the assignment are not all of the same weight. The marks for each question are shown on the sheet.

For all questions, and particularly for Questions 2 and 3, you should show your working. Some marks will be given for 'correct working or approach' even when the final answer is not correct. For Questions 2 and 4 the purpose is to understand and use the algorithms and constructions discussed in lectures. Marks will not be given for answers obtained by other means.

You should try all the questions and hand in what you have done, even if you are not able to complete some of the questions. Some marks will usually be given for correct parts of incomplete solutions. Solutions to the assignment will be discussed in a lecture slot after the assignments have been marked.

Feedback arrangements

• Solutions must be submitted in PDF format via the links on the course Moodle page. The submissions must be PDFs not jpeg, Word documents or any other format. You may hand write and then scan your solutions to obtain a PDF file.
Write your name on every page and make sure the printed version is legible.
The mark will be set to 0 if the submission is in the wrong format or not legible.

• In the event of central IT issues at the time of the deadline, inform the course lecturer.

• The assignments will be marked and a feedback grade given. Note, the feedback grades are provisional and subject to change by our External Examiners during the examination moderation process.

• General feedback for this assignment will also be posted on the course webpage on Friday 23rd November.

Learning outcomes assessed

The primary purpose of this assignment is formative. It is designed to

- provide a focus for reading and understanding the material on context free grammars, derivations, FIRST and FOLLOW sets, and on **rdp** syntax
- test your ability to use the left recursion removal technique
- test your ability to compute first and follow sets and to construct recursive descent parsers
- provide you with a way of assessing for yourselves the degree to which you have understood the material

If you do not know how to do a question, begin by reading the corresponding text in the notes and working through related examples discussed in lectures. Please feel free to make an appointment to talk to me about things you don't understand, and remember all solutions must be the submitter's own work. If someone asks you for help you may discuss the material and examples given in class but you should not give any form of help that you would not expect the course lecturer to give.