Stefan Knott
CSCI 3155 Lab 2

2)

a)

A ::= A(1) & A(2) | V
V ::= a | b

We can rewrite the grammar using the following judgement forms. (e denotes existence within a set)

		Parts of the judgement
$\frac{\text{A(1) e AObjects, A(2) e AObjects}}{\text{A(1) \& A(2) e AObjects}}$	←	premise, may be left empty for axioms
	←	what follows

$$\frac{\text{V e VObjects}}{\text{V e AObjects}}\ ,\ \frac{}{\text{a e VObjects}}\ ,\ \frac{}{\text{b e Vobjects}}$$

b)

Show the last grammar is ambiguous.
For example string aba show there are 2 possible parse trees

```
1)      A                    2)      A
       / | \                        / | \
    A    &  A                    A  &  A
   / |\       V                  V     / | \
  A & A      a                   a     A & A
   V   V                                V    V
   a    b                               b    a
```

c) The grammar forms strings consisting of 1 or more As (A+), or 0 or more Bs (B*), or 1 or more Cs (C+).

d) 1) S => AaBb => baab
2) not in the grammar
3) not in the grammar
4) S => AaBb => AbaBb => bbaBb => bbaab

e)
1) S => aScB => abcB => abcd
2) not in the grammar
3) not in the grammar
4) not in the grammar
5) s => aScB => aAcB => accB => accA => accc

3)

a)

I) This grammar forms strings which may have a large left subtree of operations strung together in a given parse tree. This left tree would then be used in a final evaluation with the original operand. This grammar can also just represent an operand value.

II) This grammar forms similar expressions, however now the right subtree of the parse tree may be arbitrarily long.

b) Example from scala REPL:

scala> 2 – 1 << 1
res0: Int2 = 2
– This shows that subtraction has a higher precedence than the binary left shift operator
(2 – 1) << 1 = 2 2 – (1 << 1) = 0

c) BNF grammar for doubles

S ::= fraction | fraction exp
fraction ::= posnum | negnum
exp: E sign num
posnum = num . dec (where "." is a decimal point)
negnum = - num . dec (where "." is a decimal point)
num ::= num b | a
a ::= {1,2,3,4,5,6,7,8,9}
b ::= {0,1,2,3,4,5,6,7,8,9}
dec ::= dec c | c
c ::= {0,1,2,3,4,5,6,7,8,9}