CMPT 383
Quiz #4
October 11, 2005

1) Prove that the following grammar is ambiguous: **(3 marks)**

```
<S>   ::=  <A>
<A>   ::=  <A>+<A>|<id>
<id>  ::= a | b | c
```

Any valid string with more than 3 terminals can be parsed in at least 2 different ways, i.e. a+b+c
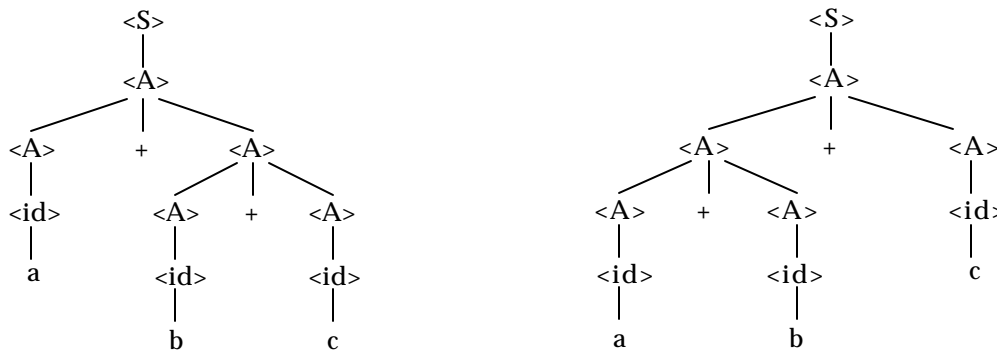
2) Convert the following EBNF to BNF: **(4 marks)**

```
<S> ::=  <A>{b<A>}
<A> ::=  a[b]<A>
```

BNF:

<S> ::= <A> | <A><B>

<B> ::= b<A> | b<A><B>

<A> ::= ab<A> | a<A>

3) Consider the following incomplete attribute grammar with nonterminals *A, B, C,* and terminals *d* and *e.* The start symbol is *A.* The attributes are assigned to these grammar symbols as indicated by the table below.

	A	B	C	d	e
synthesized	*abel, boole*	*cantor*	*descartes*	*(none)*	*(none)*
inherited	*(none)*	*euler*	*fermat*	*gauss*	*(none)*

The grammar has the following 4 productions labeled *p, q, r*, and *s*.

```
p: <A>   ::=  <B>d
q: <B>   ::=  <C><A>
r: <B₁>  ::=  <B₂>e<C>
s: <C>   ::=  d<A>
```

List the defined attribute and used attributes occurrences. **(4 marks)**

	Defined	Used
p	A.abel, A.boole, B.euler, d.gauss	B.cantor
q	B.cantor, C.fermat	B.euler, C.descartes, A.abel, A.boole
r	B1.cantor, B2.euler, C.fermat	B1.euler, B2.cantor, C.descartes
s	C.Descartes, d.gauss	C.fermat, A.abel, A.boole

4) Compute the weakest precondition for the following sequence of assignment statements and its postcondition: **(4 marks)**

```
a = 3*(2*b+a);
b = 2*a-1
{b > 5}
```

$2*a-1 > 5$ $3*(2*b+a) > 3$

$2*a > 5+1$ $2*b+a > 3/3$

$a > 6/2$ $a > 1-2*b$ or $b > (1-a)/2$

$a>3$ Weakest Precondition: $\{a > 1-2*b\}$ or $\{b > (1-a)/2\}$

5) Perform the parwise disjointness test for the following grammar rules: **(3 marks)**

```
<A> → a<B> | b | c<B><B>
```

$FIRST_1(A) = \{a\}$

$FIRST_2(A) = \{b\}$

$FIRST_3(A) = \{c\}$

$FIRST_1(A) \cap FIRST_2(A) \cap FIRST_3(A) = \varnothing$

```
<B> → a<B> | b<A> | a<B>b
```

$FIRST_1(B) = \{a\}$

$FIRST_2(B) = \{b\}$

$FIRST_3(B) = \{a\}$

$FIRST_1(B) \cap FIRST_3(B) = \{a\} \neq \varnothing$

6) Assume the following Ada program was compiled and executed using static scoping rules. What value of X is printed in procedure Sub1? Under dynamic scoping rules, what value of X is printed in procedure Sub1? **(2 marks)**

```
procedure Main is
    X : Integer;
    procedure Sub1 is
        begin              -- of Sub1
        Put(X);
        end;               -- of Sub1
    procedure Sub2 is
        X : Integer;
        begin              -- of Sub2
        X := 10;
        Sub1
```

```
        end;          -- of Sub2
    begin             -- of Main
    X := 5;
    Sub2
    end;              -- of Main
```

Dynamic Scope: X=10
Static Scope: X=5