Programming:

CSE 301 and 601:

1. Write a program to generate 3-address code for programs defined by the grammar below. Feel free to use the yacc file from the solutions to assignment 2. +, -, *, /, =, [], if, ifFalse, goto, !, <, >, <=, >=, ==, !=, param, call and ret are the allowed operators.

```
class Program { <field_decl>* <method_decl>* }
<field decl> -> <type> (<id> | <id> [ <int literal> ] ) ( , <id> | <id> [ <int literal> ] ) *;
<field decl> -> <type> <id> = teral>;
<method_decl> -> ( <type> | void ) <id> ( ((<type> <id>) ( , <type> <id>)*)? ) <block>
<blook> -> { <var decl>* <statement>* }
<var decl> -> <type> <id> ( , <id>)*;
<type> -> int | boolean
<statement> -> <location> <assign op> <expr> ;
<statement> -> <method call>;
<statement> -> if ( <expr> ) <block> ( else <block> )?
<statement> -> for <id> = <expr> , <expr> <block>
<statement> -> return ( <expr> )?;
<statement> -> break;
<statement> -> continue;
<statement> -> <block>
<assign op> -> =
<assign_op> -> +=
<assign op> -> -=
<method call> -> <method name> ( (<expr> ( , <expr> )*)? )
<method_call> -> callout ( <string_literal> ( , <callout_arg> )* )
<method name> -> <id>
<location> -> <id>
<location> -> <id>[ <expr> ]
<expr> -> <location>
<expr> -> <method_call>
<expr> -> <literal>
<expr> -> <expr> <bin op> <expr>
<expr> -> - <expr>
<expr> -> ! <expr>
<expr> -> ( <expr> )
<callout_arg> -> <expr> | <string_literal>
<br/><bin_op> -> <arith_op> | <rel_op> | <eq_op> | <cond_op>
<arith op> -> + | - | * | / | %
<rel_op> -> < | > | <= | >=
<eq_op> -> == | !=
<cond op> -> && | ||
<literal> -> <int_literal> | <char_literal> | <bool_literal>
```

```
<id> -> <alpha> <alpha_num>*
  <alpha> -> [a-zA-Z_]
  <alpha_num> -> <alpha> | <digit>
  <digit> -> [0-9]
  <hex_digit> -> <digit> | [a-fA-F]
  <int_literal> -> <decimal_literal> | <hex_literal>
  <decimal_literal> -> <digit> <digit>*
  <hex_literal> -> 0x <hex_digit> <hex_digit>*
  <bool_literal> -> true | false
  <char_literal> -> "<char>"
  <string_literal> -> "<char>""
```

(25)