Adding Actions in ANTLR

automatically adding actions

- Just like ANTLR makes it easy to convert a CFG into a parser, it makes it easy to automatically insert actions into the parser
  - Define new data to return from parse functions
  - Define how to construct that data out of the return values of right-hand-sides

```
program : var_decls funcs

...

var_decls : var_decl var_decls
          | empty

var_decl : var_type id ';'

var_type : 'int'
```

ANTLR

```
program : var_decls funcs

...

var_decls : var_decl var_decls
          | empty

var_decl : var_type id ‘;’

var_type : ‘int’
```

ANTLR

```
@members {
  SymbolTable st = ...
}
```

Add fields to parser to access during actions

```
program : var_decls funcs

...

var_decls : var_decl var_decls
          | empty

var_decl : var_type id ‘;’

var_type : ‘int’
```

ANTLR

```
@members {
  SymbolTable st = ...
}

program : var_decls funcs

...

var_decls : var_decl var_decls
          | empty

var_decl returns [VarDecl v] :
    var_type id ';' {$v = new VarDecl("int", $id.txt);}

var_type : 'int'
```

Create variable to return from parse function

Write code to construct what to return using results of RHS rules

ANTLR

```
@members {
  SymbolTable st = ...
}

program : var_decls funcs

...

var_decls returns [List<VarDecl> lv] :
        var_decl var_decls
            {$var_decls.lv.add(0, $var_decl.v);
             $lv = $var_decls.lv}
      | empty {$lv = new List<VarDecl>()}

var_decl returns [VarDecl v] :
    var_type id ';' {$v = new VarDecl("int", $id.txt);}

var_type : 'int'
```

Write code to construct what to return using results of RHS rules

next: symbol tables