

**Homework #5**

<b>Name</b>	<b>Files</b>	<b>Author</b>
High-level Intermediate Representation (HIR)	HIR.C/HIR.H, HIRVisitor.H	Casey, Abbott, Veluppillai
	Added support for string type, resolved some bugs, and added some new support methods.	
Elaboration	HIRTypeChecker.C/H	Veluppillai
	Resolved issue where certain binary operators (relation, equality, logical) now result in bool type.	
Evaluation	HIREvaluationVisitor.C/H	Abbott, Casey
	Added scoped environment support for conditional blocks, parameters, and function bodies. Resolved bugs with existing evaluation system, and added beginning support for function calls.	

**S-Expression Schema**

Program	( Variable   Function )*
Function	(" "function" Type FunctionName ParamList [ Compound ] ")
Type	(" "identifier" BasicTypes ")
FunctionName	Identifier   Operator
Identifier	(" "identifier" [a-zA-z_][a-zA-Z_0-9]* ")
Operator	(" "operator" FuncOp ")
FuncOp	"=="   "<"   "+"   "-"   "*"   "/"   "%"
ParamList	(" "Param* ")
Param	(" "parameter" Type [ Identifier ] ")
Compound	(" "compound" Statement* ")
OpExpr	(" "UnaryExpr   BinaryExpr ")
UnaryExpr	"unary_expr" UnaryOp Expression
BinaryExpr	"binary_expr" BinaryOp Expression Expression

UnaryOp	"."   "-"   "!"
BinaryOp	"=="   "="   "!="   ">"   "<"   ">="   "<="   "or"   "and"   "+"   "-"   "*"   "/"   "%"   "."
Expression	Literal   Identifier   OpExpr   Call   Field
Literal	(" "literal" LitVals ")
LitVals	"true"   "false"   ([0-9]+)   ([0-9]+\.[0-9]*)
BasicTypes	"void"   "bool"   "byte"   "int"   "double"   "char"   "string"
Statement	(" [ Label ] ( Expression   Control   Compound   ) ")
Label	"labelled" Identifier
Control	Conditional   While   Switch   Break   Goto
Conditional	"if" Expression Statement [ Statement ]
While	"while" Expression Statement
Switch	"switch" Expression Case*
Case	(" "case" Literal Statement* Break ")
Break	(" "break" "break" ")
Variable	(" "variable" Type Identifier [ Expression ] ")
Call	(" "call" Expression* ")
Field	(" "binary_expr" "." ( Identifier   OpExpr   Call   Field ) {2} ")
Goto	(" "unary_expr" "goto" Identifier ")