

VariableTable

```
-variable map: std::map<std::string, int64_t>
+push(name:std::string,value:int64_t): void
+getVariableValue(name:std::string): int
+FindVariable(name:std::string): bool
```

Visitor

```
+visitOperator(opNode:Operator*): virtual void
+visitNumber(numNode:Number*): virtual void
+visitVariable(varNode:Variable*): virtual void
+visitRelBool(relBoolNode:RelBoolOperator*): virtual void
+visitConstBool(constBoolNode:ConstBool*): virtual void
+visitBool(boolNode:BoolOperator*): virtual void
+visitPrint(printNode:StmtPrint*): virtual void
+visitSet(setNode:StmtSet*): virtual void
+visitInput(inputNode:StmtInput*): virtual void
+visitWhile(whileNode:StmtWhile*): virtual void
+visitIf(ifNode:StmtIf*): virtual void
+visitBlock(blockNode:Block*): virtual void
+visitProgram(programNode:Program*): virtual void
```

EvaluationVisitor

```
-accumulator: std::vector<int64_t>
-VpTable: VariableTables
+visitOperator(opNode:Operator*): void
+visitNumber(numNode:Number*): void
+visitVariable(varNode:Variable*): void
+visitRelBool(relBoolNode:RelBoolOperator*): void
+visitConstBool(constBoolNode:ConstBool*): void
+visitBool(boolNode:BoolOperator*): void
+visitPrint(printNode:StmtPrint*): void
+visitSet(setNode:StmtSet*): void
+visitInput(inputNode:StmtInput*): void
+visitWhile(whileNode:StmtWhile*): void
+visitIf(ifNode:StmtIf*): void
+visitBlock(blockNode:Block*): void
+visitProgram(programNode:Program*): void
```

PrintVisitor

```
+visitOperator(opNode:Operator*): void
+visitNumber(numNode:Number*): void
+visitVariable(varNode:Variable*): void
+visitRelBool(relBoolNode:RelBoolOperator*): void
+visitConstBool(constBoolNode:ConstBool*): void
+visitBool(boolNode:BoolOperator*): void
+visitPrint(printNode:StmtPrint*): void
+visitSet(setNode:StmtSet*): void
+visitInput(inputNode:StmtInput*): void
+visitWhile(whileNode:StmtWhile*): void
+visitIf(ifNode:StmtIf*): void
+visitBlock(blockNode:Block*): void
+visitProgram(programNode:Program*): void
```

Token

```
+tag: int
+word: std::string
+idWord[]: const char*
```

Tokenizer

```
-tokenizeInputFile(InFile:std::ifstream,
InTokens:std::vector<Token>): void
+operator()(InFile:ifstream&): std::vector<Token>
```

BlockManager

```
-allocated: std::vector<Block*>
+makeBlock(list:std::vector<Statement*>): Block*
+ClearMemory(): void
```

Block

```
-stmt_list: std::vector<Statement*>
+getSize(): size_t
+getBegin(): std::vector<Statement*>::const_iterator
+getEnd(): std::vector<Statement*>::const_iterator
+accept(v:Visitor*): void
```

Program

```
-block: Block*
+getBlock(): Block*
+accept(v:Visitor*): void
```

StmtManager

```
-allocated: std::vector<Statement*>
+makeStmtPrint(expr:NumExpr*): Statement*
+makeStmtSet(var:Variable*,expr:NumExpr*): Statement*
+makeStmtInput(var:Variable*): Statement*
+makeStmtWhile(expr:BoolExpr*,b:Block*): Statement*
+makeStmtIf(i:Block*,z:Block*,b:expr:BoolExpr*): Statement*
+ClearMemory(): void
```

Statement

```
+Statement(): virtual
+accept(v:Visitor*): virtual void
```

Print

```
-expr: NumExpr*
+getExpr(): NumExpr*
+accept(v:Visitor*): void
```

Set

```
-var: Variable
+getVar(): Variable
+getExpr(): NumExpr*
+accept(v:Visitor*): void
```

Input

```
-var: variable
+getVariable(): variable
+accept(v:Visitor*): void
```

While

```
-bool expr: BoolExpr*
-stmt block: Block
+getBoolExpr(): BoolExpr*
+getStmtBlock(): Block*
+accept(v:Visitor*): void
```

IfStmt

```
-stmt block1: Block
-stmt block2: Block
-bool expr: BoolExpr*
+getRight(): Block*
+getLeft(): Block*
+getExpr(): BoolExpr*
+accept(v:Visitor*): void
```

NumExprManager

```
-num allocated: std::vector<NumExpr*>
+makeNumber(value:int64_t): NumExpr*
+makeOperator(op:Operator::OpCode,i:NumExpr*,
z:NumExpr*): NumExpr*
+makeVariable(var:std::string): NumExpr*
+ClearMemory(): void
```

NumExpr

```
+accept(v:Visitor*): virtual void
+NumExpr(): virtual
```

Operator

```
-op: OpCode
-left: Expression*
-right: Expression*
+OpCode: enum = ADD, SUB, MUL, DIV
+getOp(): OpCode
+getLeft(): Expression*
+getRight(): Expression*
+accept(v:Visitor*): void
```

Number

```
-value: int64_t
+get_value(): int64_t
+accept(v:Visitor*): void
```

Variable

```
-name: std::string
-value: NumExpr*
+accept(v:Visitor*): void
+getName(): std::string
+getValue(): NumExpr*
```

BoolExprManager

```
-allocated: std::vector<BoolExpr*>
+makeRelBool(op:RelBoolOperator::RelOpCode,
l:NumExpr*,r:NumExpr*): BoolExpr*
+makeConstBool(value:int): BoolExpr*
+makeBoolOp(op:BoolOperator::OpCodeBool,
expr_bool1:BoolExpr*,expr_bool2:BoolExpr*): BoolExpr*
+ClearMemory(): void
```

BoolExpr

```
+accept(v:Visitor*): virtual void
+BoolExpr(): virtual
```

RelOp

```
-Rel_Boolean: RelOpCode
-Rel_left: NumExpr*
-Rel_right: NumExpr*
+RelOpCode: enum = GT, LT, EQ
+getRelOp(): RelOpCode
+getLeftOp(): NumExpr*
+getRightOp(): NumExpr*
+accept(v:Visitor*): void
```

ConstOp

```
-value: int
+getValue(): int
```

BoolOp

```
-BoolOp: OpCodeBool
-left: BoolExpr*
-right: BoolExpr*
+OpCodeBool: enum = AND, OR, NOT
+getBoolOp(): OpCodeBool
+getLeftOp(): BoolExpr*
+getRightOp(): BoolExpr*
```

Parser

```
-streamEnd: std::vector<Token*>::const_iterator
-allocated_stmt: std::vector<Statement*>
-check_parenthesis: int
-num_em: NumExprManager*
-bool_em: BoolExprManager*
-stmt_em: StmtManager*
-block_em: BlockManager*
+recursiveBlockParse(tokenIdx:(std::vector<Token*>::const_iterator): Block*
+recursiveStmtParse(tokenIdx:(std::vector<Token*>::const_iterator): Statement*
+recursiveBoolParse(tokenIdx:(std::vector<Token*>::const_iterator): BoolExpr*
+recursiveNumParse(tokenIdx:(std::vector<Token*>::const_iterator): NumExpr*
+safe_next(itr:(std::vector<Token*>::const_iterator): void
+check_overflow(itr:(std::vector<Token*>::const_iterator): void
+operator()(tokenStream:const std::vector<Token*>): Program*
```