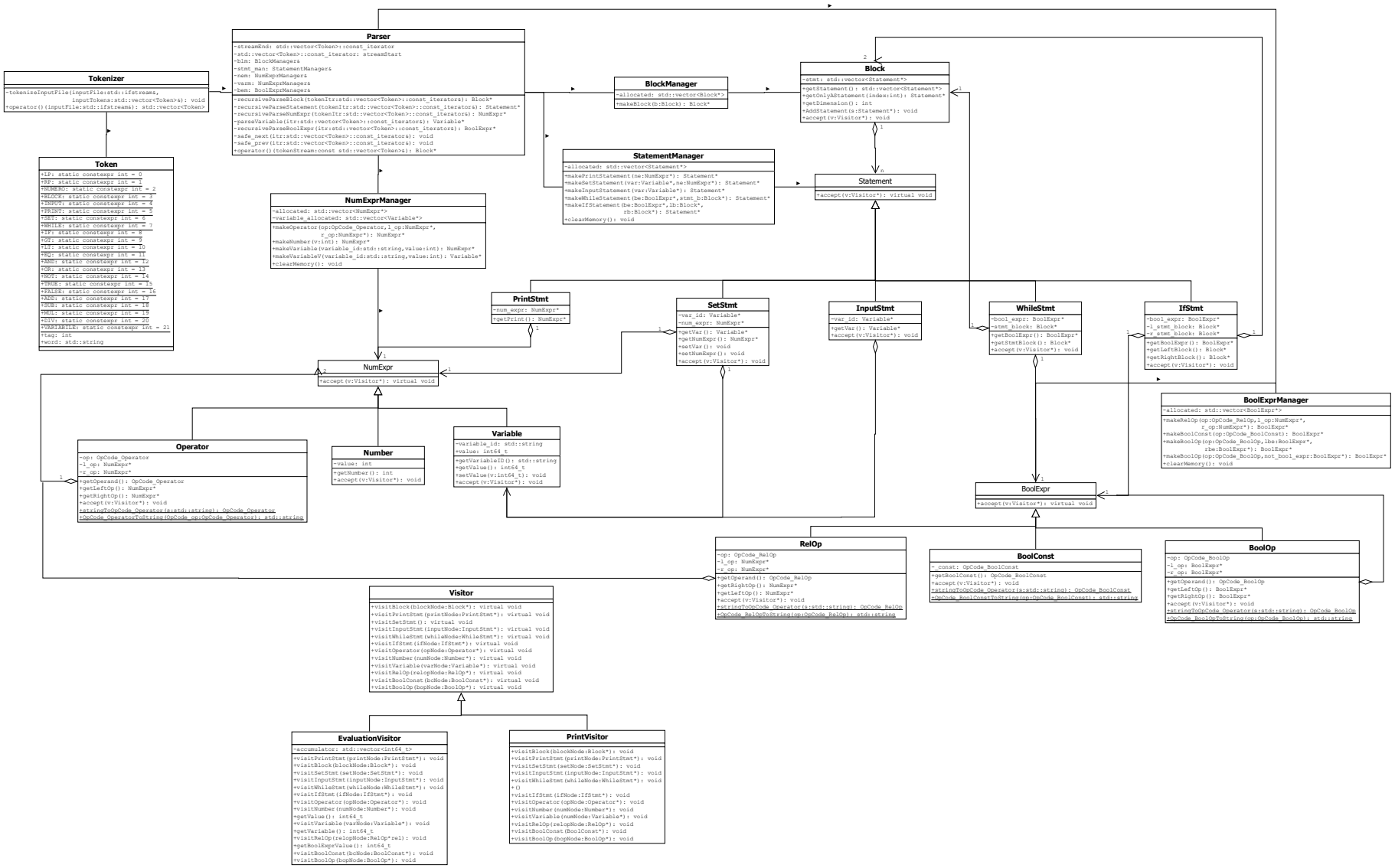
Tokenizer
-tokenizeInputFile(inputFile:std::ifstream&,
inputTokens:std::vector<Token>&): void
+operator()(inputFile:std::ifstream&): std::vector<Token>
Parser
-streamEnd: std::vector<Token>::const_iterator
-std::vector<Token>::const_iterator: streamStart
-blm: BlockManager&
-stmt_man: StatementManager&
-nem: NumExprManager&
-varm: NumExprManager&
-bem: BoolExprManager&
-recursiveParseBlock(tokenItr:std::vector<Token>::const_iterator&): Block*
-recursiveParseStatement(tokenItr:std::vector<Token>::const_iterator&): Statement*
-recursiveParseNumExpr(tokenItr:std::vector<Token>::const_iterator&): NumExpr*
-parseVariable(itr:std::vector<Token>::const_iterator&): Variable*
-recursiveParseBoolExpr(itr:std::vector<Token>::const_iterator&): BoolExpr*
-safe_next(itr:std::vector<Token>::const_iterator&): void
-safe_prev(itr:std::vector<Token>::const_iterator&): void
+operator()(tokenStream:const std::vector<Token>&): Block*
BlockManager
-allocated: std::vector<Block*>
+makeBlock(b:Block): Block*
Block
-stmt: std::vector<Statement*>
+getStatement(): std::vector<Statement*>
+getOnlyAStatement(index:int): Statement*
+getDimension(): int
+AddStatement(s:Statement*): void
+accept(v:Visitor*): void
Token
+LP: static constexpr int = 0
+RP: static constexpr int = 1
+NUMERO: static constexpr int = 2
+BLOCK: static constexpr int = 3
+INPUT: static constexpr int = 4
+PRINT: static constexpr int = 5
+SET: static constexpr int = 6
+WHILE: static constexpr int = 7
+IF: static constexpr int = 8
+GT: static constexpr int = 9
+LT: static constexpr int = 10
+EQ: static constexpr int = 11
+AND: static constexpr int = 12
+OR: static constexpr int = 13
+NOT: static constexpr int = 14
+TRUE: static constexpr int = 15
+FALSE: static constexpr int = 16
+ADD: static constexpr int = 17
+SUB: static constexpr int = 18
+MUL: static constexpr int = 19
+DIV: static constexpr int = 20
+VARIABLE: static constexpr int = 21
+tag: int
+word: std::string
NumExprManager
-allocated: std::vector<NumExpr*>
-variable_allocated: std::vector<Variable*>
+makeOperator(op:OpCode_Operator,l_op:NumExpr*,
r_op:NumExpr*): NumExpr*
+makeNumber(v:int): NumExpr*
+makeVariable(variable_id:std::string,value:int): NumExpr*
+makeVariableV(variable_id:std::string,value:int): Variable*
+clearMemory(): void
StatementManager
-allocated: std::vector<Statement*>
+makePrintStatement(ne:NumExpr*): Statement*
+makeSetStatement(var:Variable*,ne:NumExpr*): Statement*
+makeInputStatement(var:Variable*): Statement*
+makeWhileStatement(be:BoolExpr*,stmt_b:Block*): Statement*
+makeIfStatement(be:BoolExpr*,lb:Block*,
rb:Block*): Statement*
+clearMemory(): void
Statement
+accept(v:Visitor*): virtual void
PrintStmt
-num_expr: NumExpr*
+getPrint(): NumExpr*
SetStmt
-var_id: Variable*
-num_expr: NumExpr*
+getVar(): Variable*
+getNumExpr(): NumExpr*
+setVar(): void
+setNumExpr(): void
+accept(v:Visitor*): void
InputStmt
-var_id: Variable*
+getVar(): Variable*
+accept(v:Visitor*): void
WhileStmt
-bool_expr: BoolExpr*
-stmt_block: Block*
+getBoolExpr(): BoolExpr*
+getStmtBlock(): Block*
+accept(v:Visitor*): void
IfStmt
-bool_expr: BoolExpr*
-l_stmt_block: Block*
-r_stmt_block: Block*
+getBoolExpr(): BoolExpr*
+getLeftBlock(): Block*
+getRightBlock(): Block*
+accept(v:Visitor*): void
NumExpr
+accept(v:Visitor*): virtual void
BoolExprManager
-allocated: std::vector<BoolExpr*>
+makeRelOp(op:OpCode_RelOp,l_op:NumExpr*,
r_op:NumExpr*): BoolExpr*
+makeBoolConst(op:OpCode_BoolConst): BoolExpr*
+makeBoolOp(op:OpCode_BoolOp,lbe:BoolExpr*,
rbe:BoolExpr*): BoolExpr*
+makeBoolOp(op:OpCode_BoolOp,not_bool_expr:BoolExpr*): BoolExpr*
+clearMemory(): void
Operator
-op: OpCode_Operator
-l_op: NumExpr*
-r_op: NumExpr*
+getOperand(): OpCode_Operator
+getLeftOp(): NumExpr*
+getRightOp(): NumExpr*
+accept(v:Visitor*): void
+stringToOpCode_Operator(s:std::string): OpCode_Operator
+OpCode_OperatorToString(OpCode_op:OpCode_Operator): std::string
Number
-value: int
+getNumber(): int
+accept(v:Visitor*): void
Variable
-variable_id: std::string
+value: int64_t
+getVariableID(): std::string
+getValue(): int64_t
+setValue(v:int64_t): void
+accept(v:Visitor*): void
BoolExpr
+accept(v:Visitor*): virtual void
RelOp
-op: OpCode_RelOp
-l_op: NumExpr*
-r_op: NumExpr*
+getOperand(): OpCode_RelOp
+getRightOp(): NumExpr*
+getLeftOp(): NumExpr*
+accept(v:Visitor*): void
+stringToOpCode_Operator(s:std::string): OpCode_RelOp
+OpCode_RelOpToString(op:OpCode_RelOp): std::string
BoolConst
-_const: OpCode_BoolConst
+getBoolConst(): OpCode_BoolConst
+accept(v:Visitor*): void
+stringToOpCode_Operator(s:std::string): OpCode_BoolConst
+OpCode_BoolConstToString(op:OpCode_BoolConst): std::string
BoolOp
-op: OpCode_BoolOp
-l_op: BoolExpr*
-r_op: BoolExpr*
+getOperand(): OpCode_BoolOp
+getLeftOp(): BoolExpr*
+getRightOp(): BoolExpr*
+accept(v:Visitor*): void
+stringToOpCode_Operator(s:std::string): OpCode_BoolOp
+OpCode_BoolOpToString(op:OpCode_BoolOp): std::string
Visitor
+visitBlock(blockNode:Block*): virtual void
+visitPrintStmt(printNode:PrintStmt*): virtual void
+visitSetStmt(): virtual void
+visitInputStmt(inputNode:InputStmt*): virtual void
+visitWhileStmt(whileNode:WhileStmt*): virtual void
+visitIfStmt(ifNode:IfStmt*): virtual void
+visitOperator(opNode:Operator*): virtual void
+visitNumber(numNode:Number*): virtual void
+visitVariable(varNode:Variable*): virtual void
+visitRelOp(relopNode:RelOp*): virtual void
+visitBoolConst(bcNode:BoolConst*): virtual void
+visitBoolOp(bopNode:BoolOp*): virtual void
EvaluationVisitor
-accumulator: std::vector<int64_t>
+visitPrintStmt(printNode:PrintStmt*): void
+visitBlock(blockNode:Block*): void
+visitSetStmt(setNode:SetStmt*): void
+visitInputStmt(inputNode:InputStmt*): void
+visitWhileStmt(whileNode:WhileStmt*): void
+visitIfStmt(ifNode:IfStmt*): void
+visitOperator(opNode:Operator*): void
+visitNumber(numNode:Number*): void
+getValue(): int64_t
+visitVariable(varNode:Variable*): void
+getVariable(): int64_t
+visitRelOp(relopNode:RelOp*rel): void
+getBoolExprValue(): int64_t
+visitBoolConst(bcNode:BoolConst*): void
+visitBoolOp(bopNode:BoolOp*): void
PrintVisitor
+visitBlock(blockNode:Block*): void
+visitPrintStmt(printNode:PrintStmt*): void
+visitSetStmt(setNode:SetStmt*): void
+visitInputStmt(inputNode:InputStmt*): void
+visitWhileStmt(whileNode:WhileStmt*): void
+()
+visitIfStmt(ifNode:IfStmt*): void
+visitOperator(opNode:Operator*): void
+visitNumber(numNode:Number*): void
+visitVariable(numNode:Variable*): void
+visitRelOp(relopNode:RelOp*): void
+visitBoolConst(BoolConst*): void
+visitBoolOp(bopNode:BoolOp*): void