CSE431 – Translation of Computer Languages

- Type Checking

Doug Shook

Symbol Tables

- What kinds of information do we expect to find in a symbol table?

Attributes

- Each identifier will have attributes associated with it
  - Anything that the compiler knows
  - Where might the compiler get information about identifiers in the code?
- Nearly all attributes will contain a TypeDescriptor
  - TypeDescriptors contain fields describing the type

Type Checking

- Relies on the Visitor Pattern

- NodeVisitor (SemanticsVisitor): used as a base class
  - SymtabVisitor (TopDeclVisitor)
    - Invoked by NodeVisitor for *declarations*
    - Creates the symbol table
  - TypeVisitor
    - Invoked by SemanticsVisitor for *definitions*
    - Constructs TypeDescriptors

SymtabVisitor

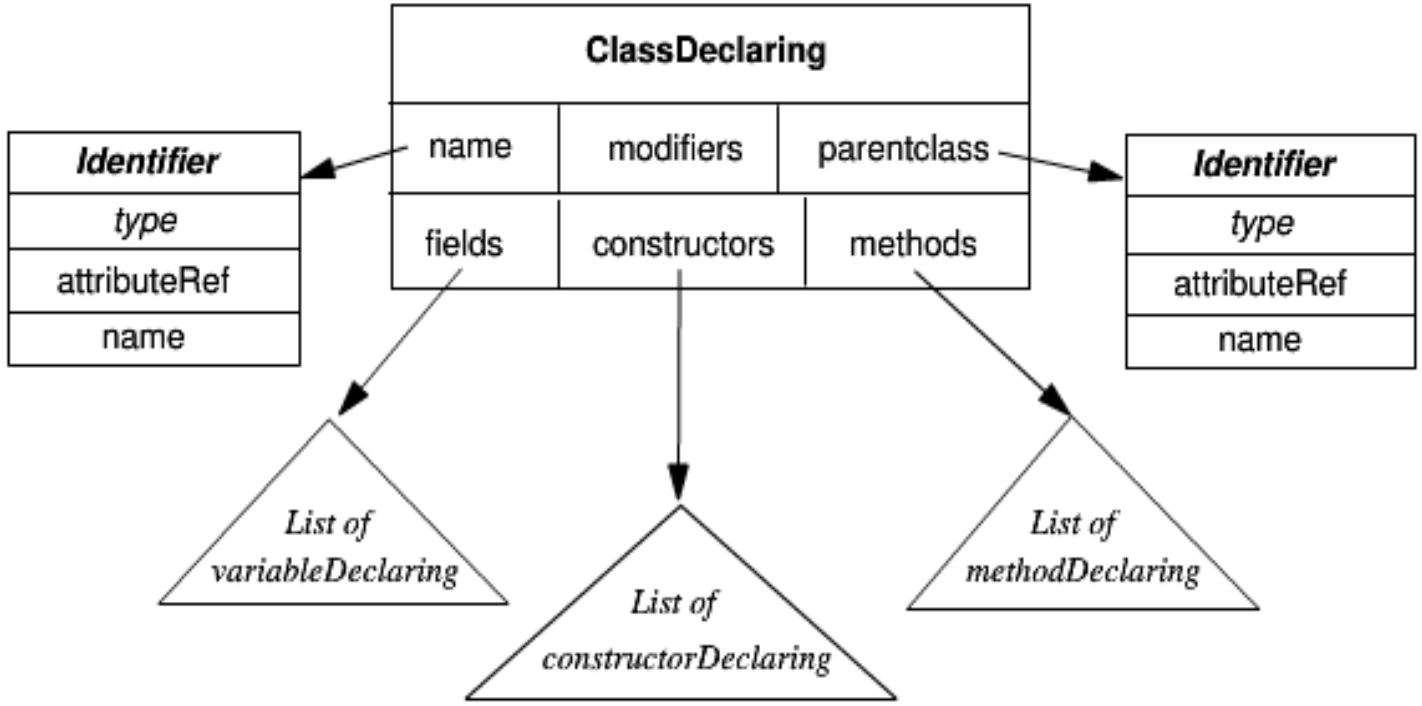
- Handles (mostly) *declarations*
  - Local Variables
    - Declarations and references
  - Classes
  - Fields
  - Methods
  - Blocks
    - Why?

SymtabVisitor

- Local Variables
  - What do we need to do upon variable declaration?
    - What could potentially go wrong?
  - What do we need to do when a variable is referenced?
    - What could potentially go wrong?

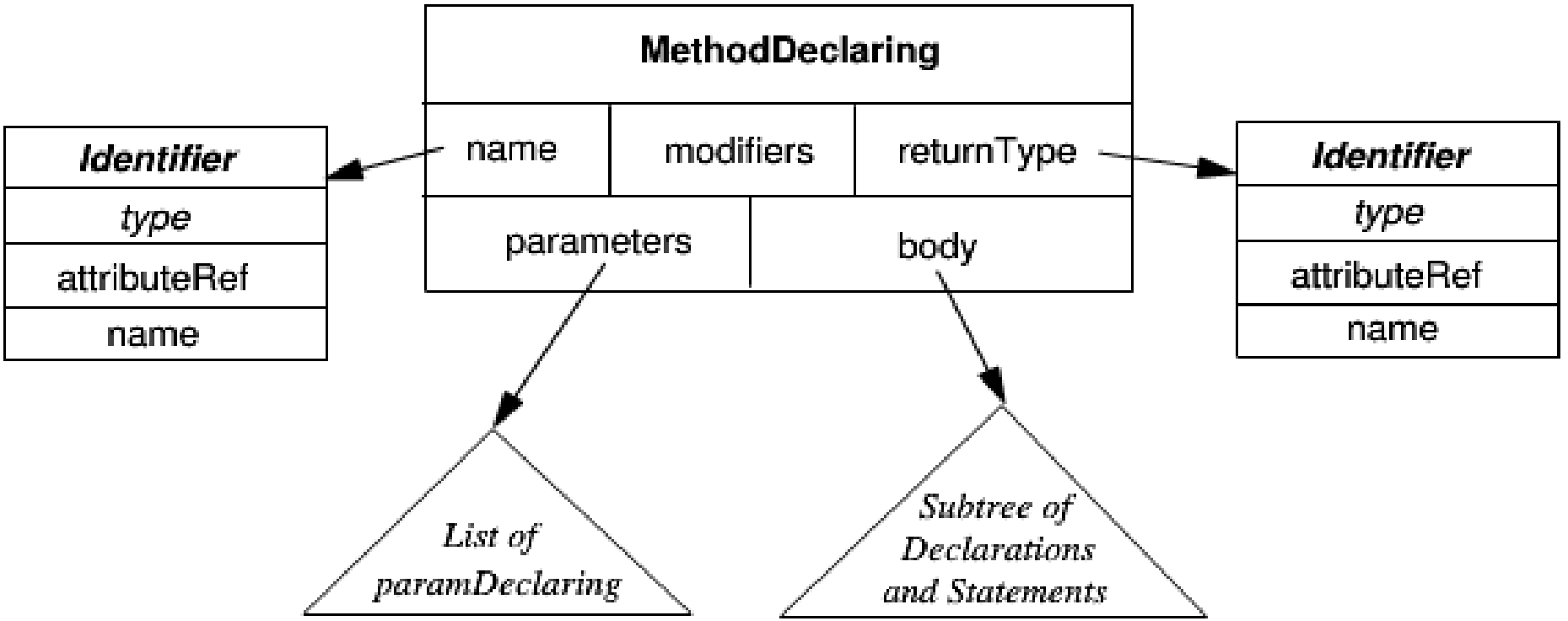
SymtabVisitor

- Classes
  - What do we need to do upon a class definition?
  - We must also pass the visitor to the class node's children
    - Why?



SymtabVisitor

- Methods
  - What do we need to do upon a method definition?
    - How should parameters be handled?
    - How should the method body be handled?



SymtabVisitor

- Fields
  - What do we need to do upon variable declaration?
    - What could potentially go wrong?
- Blocks
  - What do we need to do upon encountering a block?

SymInfo

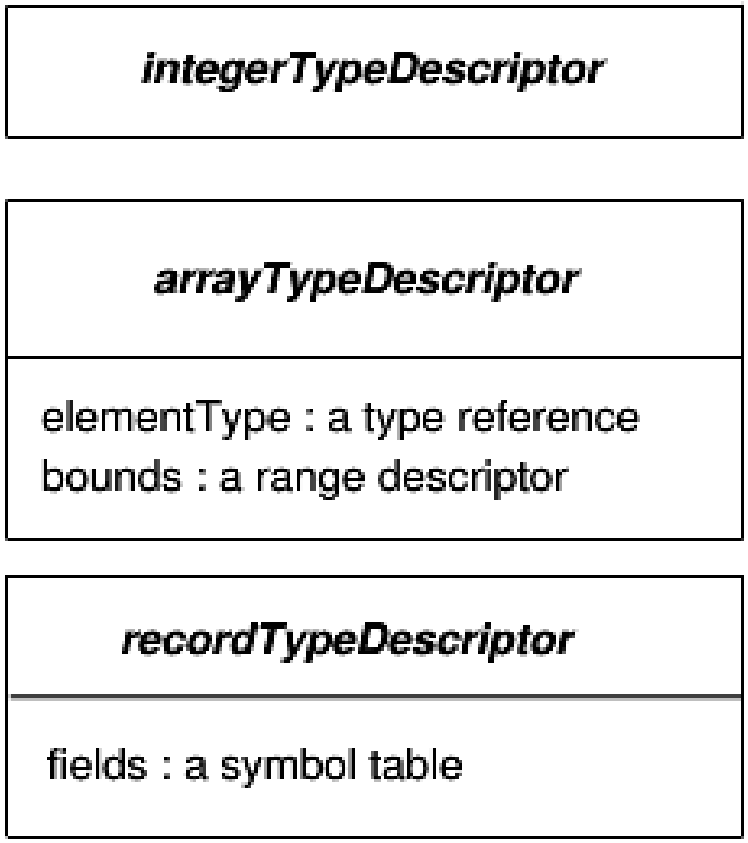
- The *identifier* blocks from the diagrams correspond to the SymInfo interface
  - Symbol table stores SymInfo objects
  - AST is decorated with SymInfo objects
  - You will need to implement this
    - Defining Node
    - Type
    - Mods

Lab 4

- Overview of assignment and code

TypeVisitor

- Primary responsibility: create TypeDescriptors



TypeVisitor

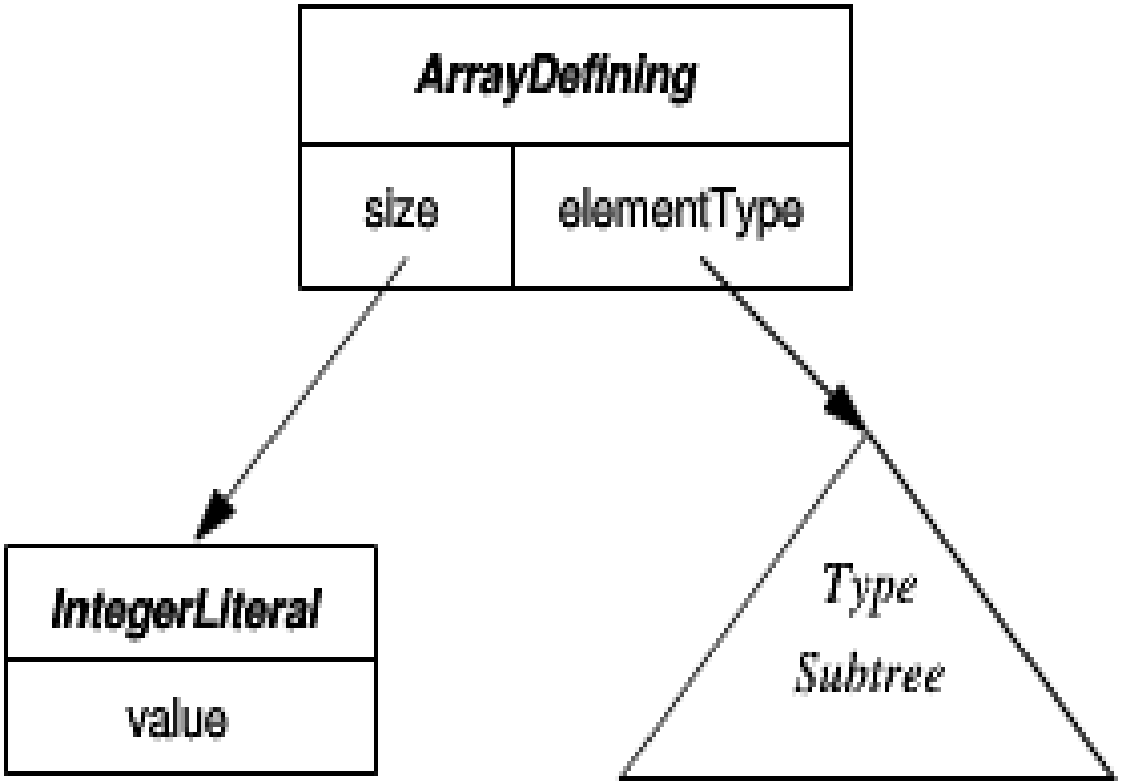
```
class TypeVisitor extends TopDeclVisitor
    procedure VISIT( Identifier id )                                    (16)
        /*  Section 8.6.2 on page 304                                   */
    end
    procedure VISIT( ArrayDefining arraydef )                           (17)
        /*  Section 8.6.5 on page 311                                   */
    end
    procedure VISIT( StructDefining structdef )                         (18)
        /*  Section 8.6.6 on page 312                                   */
    end
    procedure VISIT( EnumDefining enumdef )                             (19)
        /*  Section 8.6.7 on page 313                                   */
    end
end
```

TypeVisitor

- Where is type information stored?
- What is the first thing that should be done when processing a type identifier?

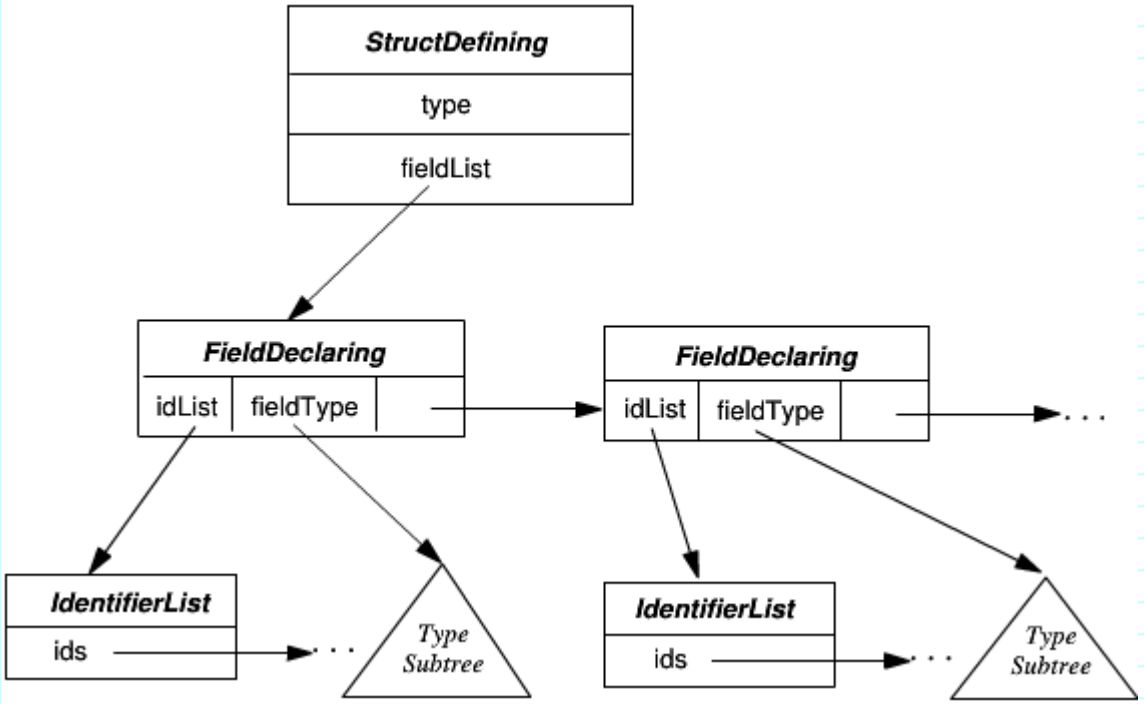
Type Definitions

- Arrays
  - Commonly: required to define array type and size
    - elementType: what is this?
    - arraySize



Type Definitions

- Structures
  - Stripped down version of a class
  - Idea: create a symbol table for the fields
    - Why is this a better idea than putting values into the global symbol table?
    - How does this compare with classes?



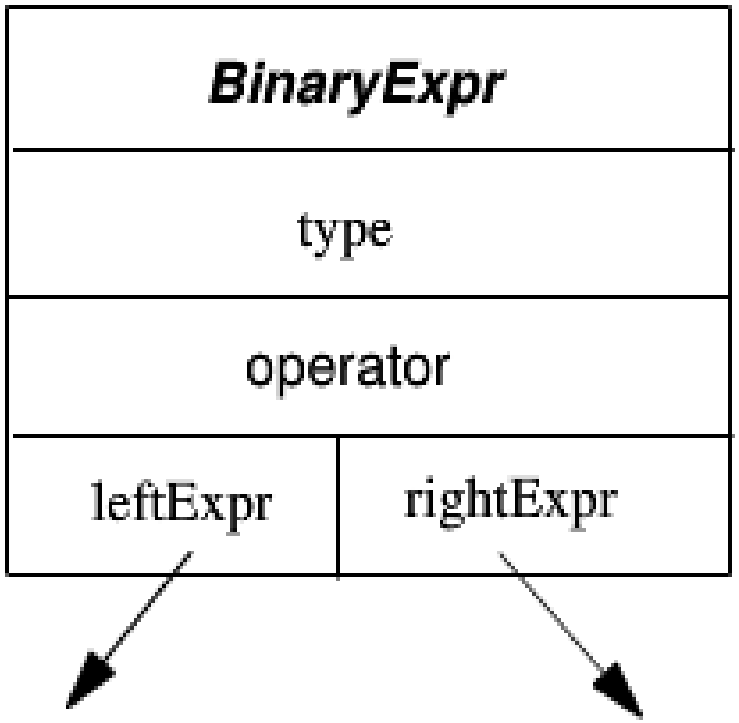
Type Checking

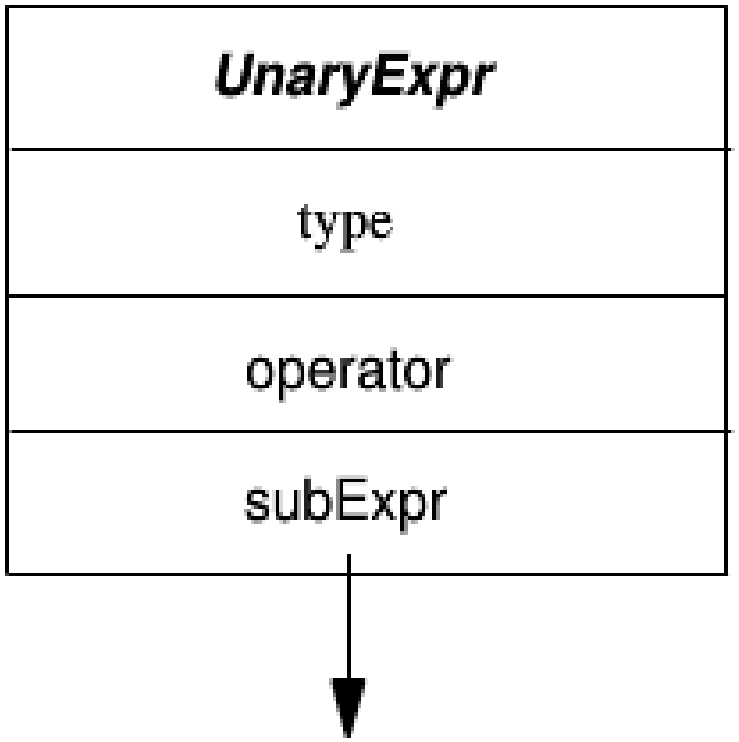
- When is it important for us to check types?

Type Checking

- Identifiers and literals:
  - Look up the type
  - Use errorType if it cannot be determined
- Assignments:
  - Need to be aware of left vs. right
    - Left is....?
    - Right is...?
  - LHSVisitor
    - Determine the type
    - Can we write here?
  - RHS will be what?

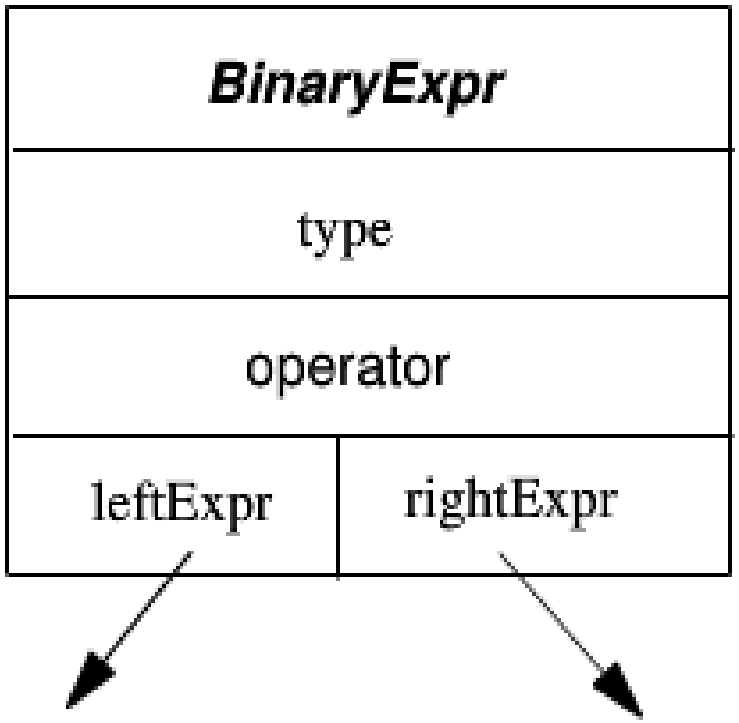
BinaryExpr

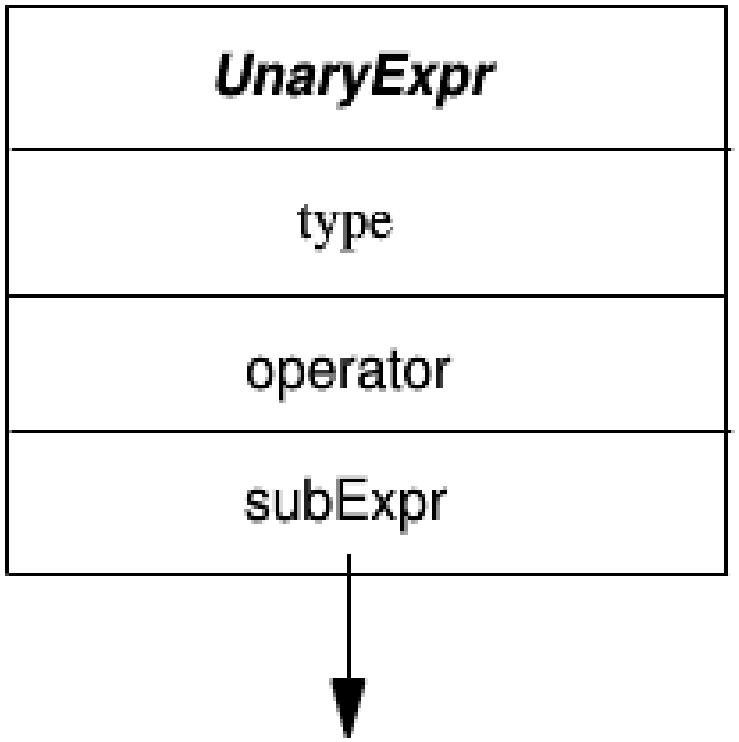




- Which of the above pieces need to be type-checked?

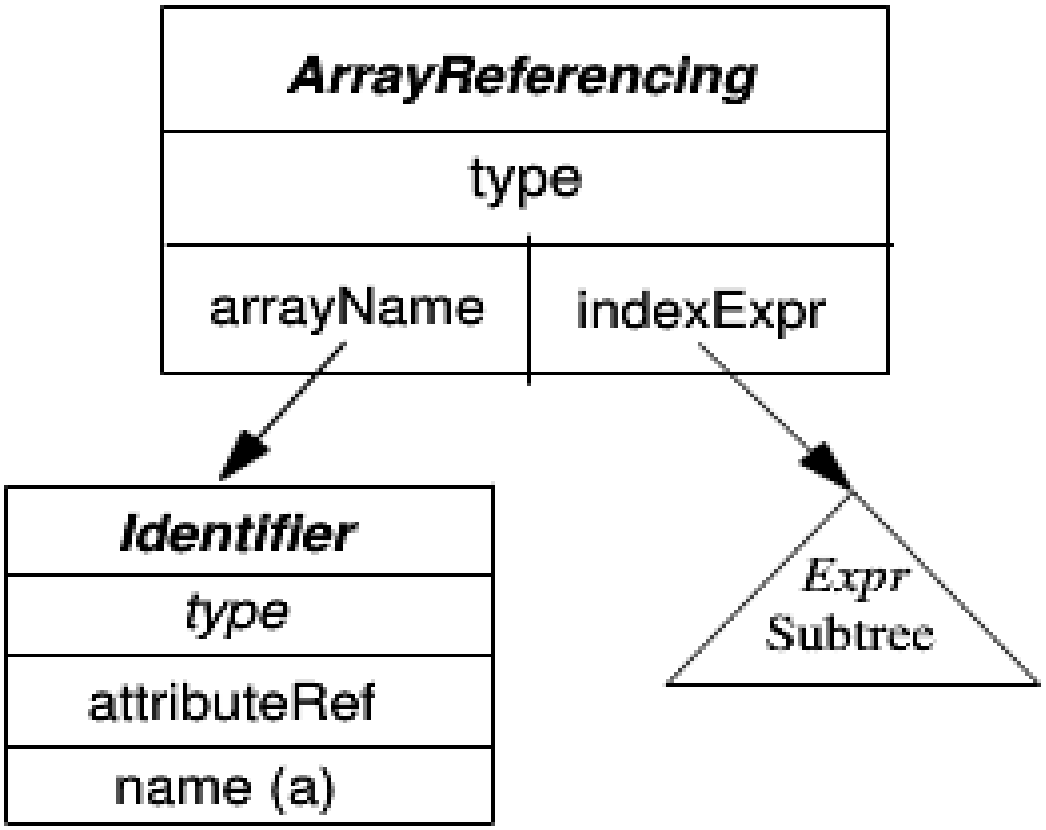
BinaryExpr





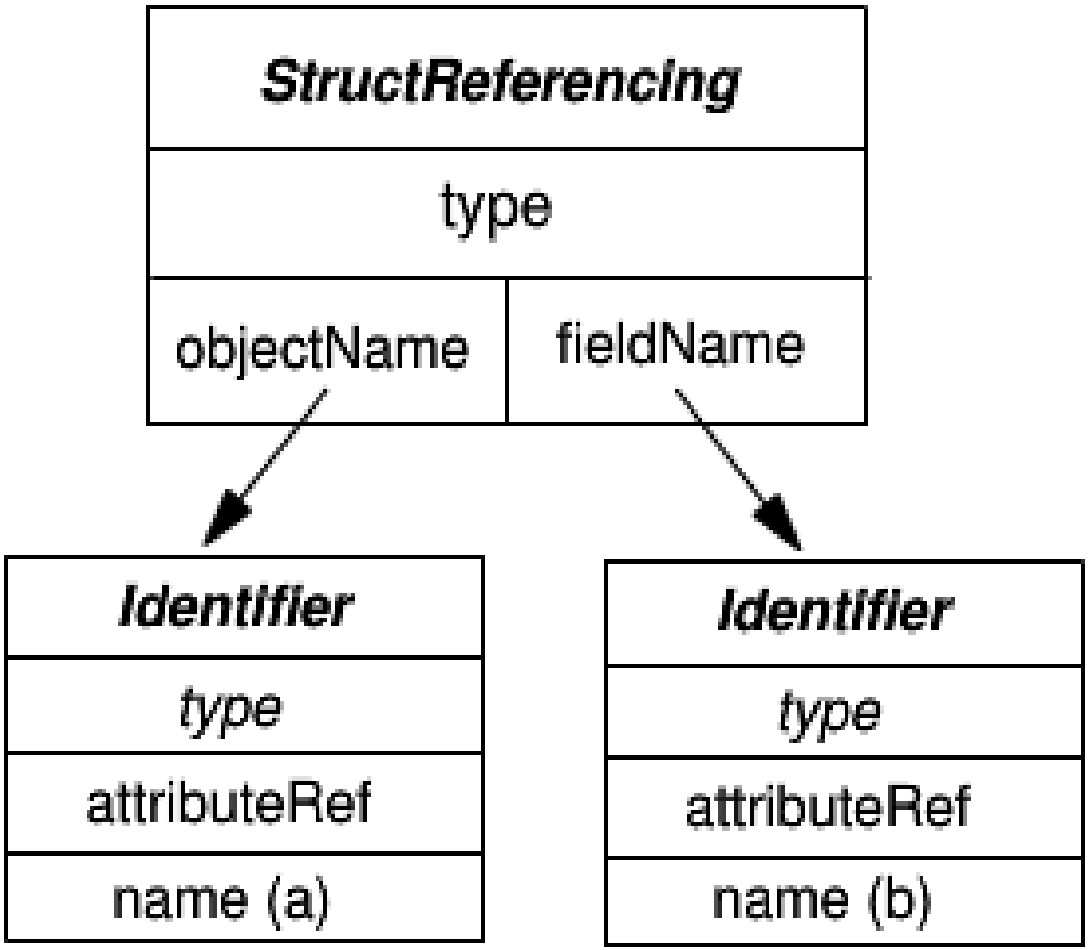
- Which of the above pieces need to be type-checked?

Arrays



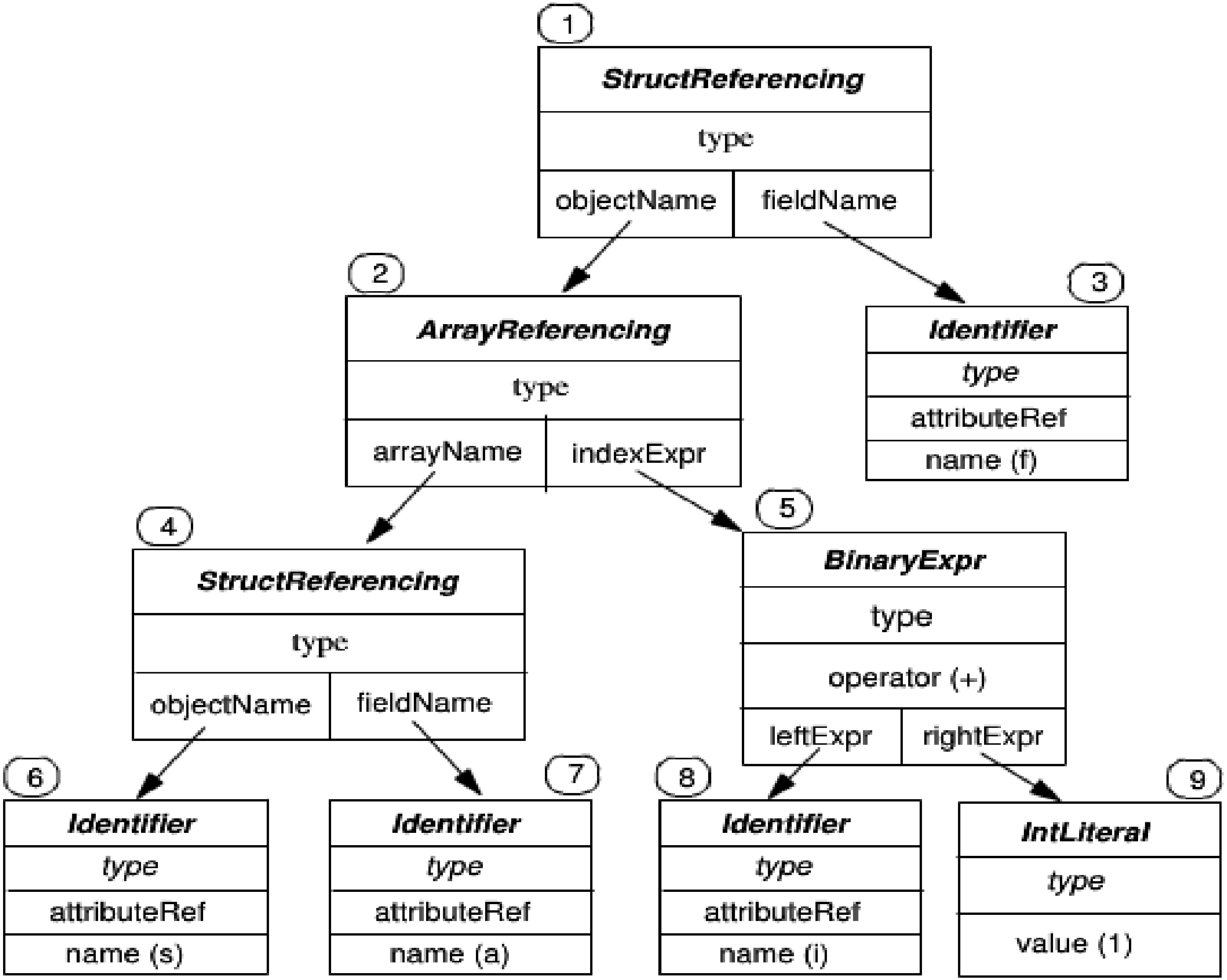
- Which of the above pieces need to be type checked?

Structs



- Which of the above pieces need to be type checked?

Example


1
StructReferencing
type
objectName
fieldName
2
ArrayReferencing
type
arrayName
indexExpr
3
Identifier
type
attributeRef
name (f)
4
StructReferencing
type
objectName
fieldName
5
BinaryExpr
type
operator (+)
leftExpr
rightExpr
6
Identifier
type
attributeRef
name (s)
7
Identifier
type
attributeRef
name (a)
8
Identifier
type
attributeRef
name (i)
9
IntLiteral
type
value (1)

Exercises

- When processing an assignment, the TypeChecking visitor must determine whether the type of the value on the right can be assigned to the variable on the left. Describe how this might be implemented.

- What should TypeChecking.visit(AssignNode) do when the right child is a method call?

- What should TypeChecking.visit(BinaryExpr) do when both children are method calls?