Plan for Today

PA7 overview

Symbol Table Implementation
– how type information is represented with SymTable and Type data structures

Type errors in MiniJava
– errors relevant to classes and arrays

Implementing classes
– dynamically allocating an object
– the “this” parameter and expression
– uses and defines of class member variables

Type implementation in the MiniJava compiler

```java
public class Type {
    public static final Type ARRAY = new Type();

    public static final Type BOOL = new Type();

    public static final Type INT = new Type();

    // class type map (key: class name, value: type)
    private static final HashMap<String, Type> classTypes
        = new HashMap<String, Type>();

    Only one instance of the type object per atomic type and class type
    – to determine if types are equal just compare references
    – Type class does not know about inheritance
```
MiniJava Types for Example

Implementing type checking for MiniJava (Classes)

**Syntax**

<table>
<thead>
<tr>
<th>New id ( )</th>
<th>NewExp</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>[LINENUM,POSNUM] Class CLASSNAME does not exist</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Exp . id ( ExpList )</th>
<th>CallExp</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>[LINENUM,POSNUM] Receiver of method call must be a class type</td>
</tr>
<tr>
<td></td>
<td>[LINENUM,POSNUM] Method METHODNAME does not exist in class type</td>
</tr>
<tr>
<td>CLASSNAME</td>
<td>Method METHODNAME requires exactly NUM arguments</td>
</tr>
<tr>
<td></td>
<td>[LINENUM,POSNUM] Invalid argument type for method METHODNAME</td>
</tr>
</tbody>
</table>
Implementing type checking for MiniJava (Arrays)

<table>
<thead>
<tr>
<th>Syntax</th>
<th>AST Node(s)</th>
</tr>
</thead>
</table>
| `new int [ Exp ]` | **NewArrayExp**  
[LINEUM,POSNUM] Invalid operand type for new array operator |
| `Exp [ Exp ]`     | **ArrayExp and ArrayAssignStatement**  
[LINEUM,POSNUM] Array reference to non-array type  
[LINEUM,POSNUM] Invalid index expression type for array reference  
[LINEUM,POSNUM] Invalid expression type assigned into array |
| `Exp . length`    | **LengthExp**  
[LINEUM,POSNUM] Operator length called on non-array type |