CMPT 432 - SPRING 2021 - DR. LABOUSEUR

Lab Seven

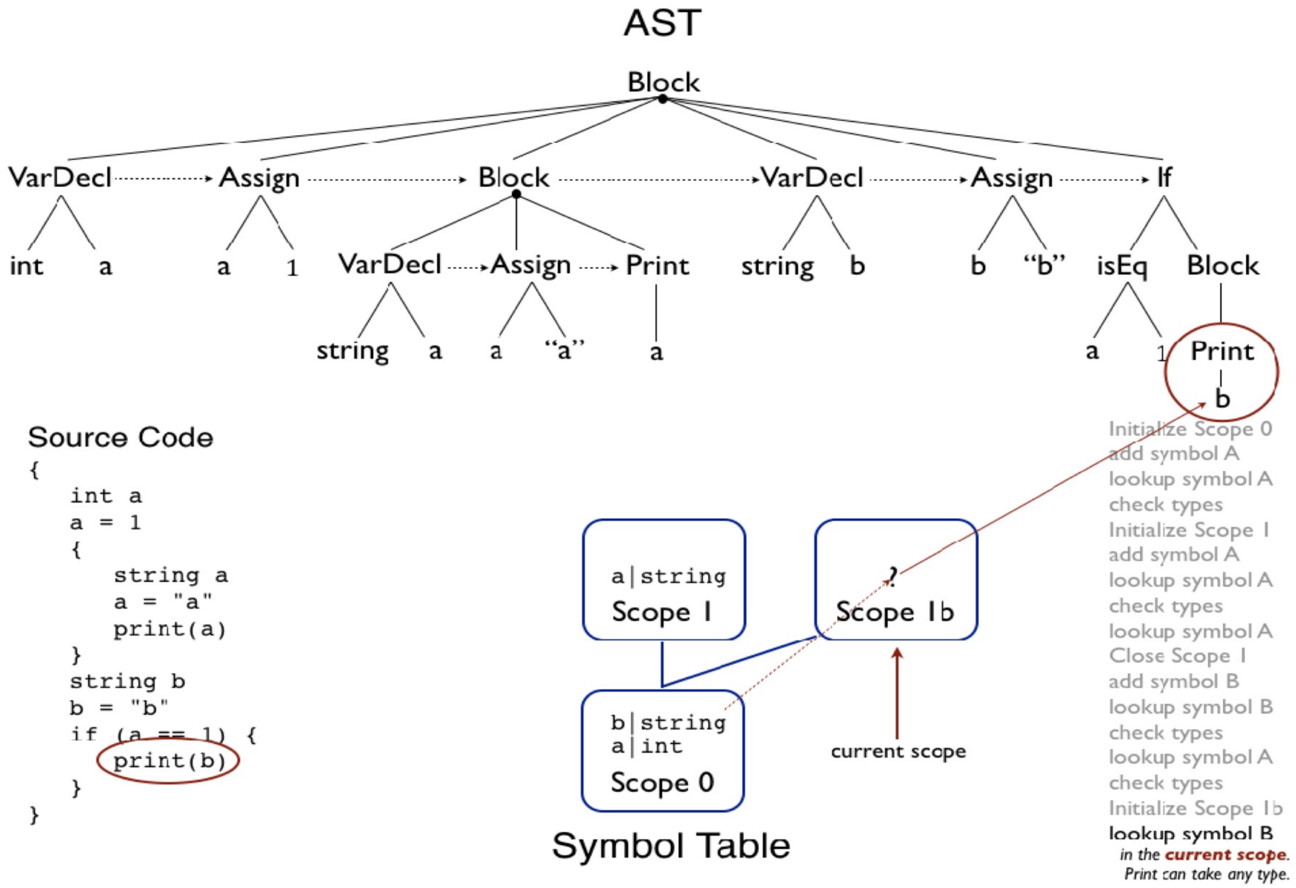
Emily Doran

Emily.Doran1@Marist.edu

April 22, 2021

Describe in detail what is happening in the diagram below.

The diagram below shows the Abstract Syntax tree and the symbol table for the source code program shown. ASTs only contain the "important" information, unlike CSTs which have everything. The symbol table contains the different scopes and the variables that are declared in each scope. This allows us to do scope checking on the programs to make sure that when a variable is being used, it has already been declared in our scope, and also helps if the same variable is declared in multiple scopes, we need to know which value to use in our current scope. The diagram describes the process of initializing each scope, and as variables are called, we look it up in the symbol table to make sure it's already been declared. For example, in the diagram, when print(b) is called, we need to check the current scope (1b) to make sure that b is declared before it is used. Since it has not been declared in this scope, we next check the "parent" scope (scope 0). Since b is declared in the parent scope, we don't need to throw an error.



Source Code

```
{
   int a
   a = 1
   {
      string a
      a = "a"
      print(a)
   }
   string b
   b = "b"
   if (a == 1) {
      print(b)
   }
}
```

Initialize Scope 0
add symbol A
lookup symbol A
check types
Initialize Scope 1
add symbol A
lookup symbol A
check types
lookup symbol A
Close Scope 1
add symbol B
lookup symbol B
check types
lookup symbol A
check types
Initialize Scope 1b
lookup symbol B
*in the **current scope**.*
Print can take any type.