

C.

Creating an Environment

or, Variables!¹

Spring 2020

¹Well, actually named constants.

Storing Values by Name

- Any reasonable language will need to have a way to refer to the values in a program by name.
- Static Properties of Variables
 - For now we are talking about
 - name → variable(..properties..),
 - not variable → value, which is a run-time property
- A "static" property is one we can figure out from the source.
- Let's look at properties, including *scope*, of some Java code.

What do we know about all these identifiers?

```
public class Foo {  
    public static int y;  
    public int z;  
    public static void main(String [] args) {  
        // args is local to main  
        Foo f = new Foo(); // f is local to main  
        int x = 1; // x is local in main  
        Foo.y = 2; // y is static throughout in Foo  
        f.z = 3; // z is known only within instances of Foo  
        { // beginning of block  
            int x = 4;  
            System.out.println(x);  
        } // end of block  
        System.out.println(x);  
    }  
}
```

- type
- scope
- memory model

Static versus Dynamic Binding

```
def main():
    x = 34
    def sub1():
        x = 56
        sub2()
    def sub2():
        print( x )

    sub1()
main()
```

dynamic: "56"
static: "34"
Which one do
you prefer?

A Slight Diversion

```
void f( int x, String y ) {  
    int total = x * scale;  
    print( "Hi, " + y );  
    check( total );  
}
```

- Bound variables
 - f, x, y, total
 - Free variables
 - scale, check
 - ("Hi, " is not a variable.)
-
- A complete program has no free variables.

Defining Environment

- For now, variables just have scope, and a value at run time.
 - (later, typing)
- We can express an environment abstractly in the form
 - $\{(s_1, v_1), \dots, (s_n, v_n)\}$
 - where s is "symbol" and v is "value".

simplified interface

```
public abstract class Env
    applyEnv(symbol) - looks up symbol in environment
    extendEnv(symbols-and-values): Env
        - adds new symbol+value to environment, returns new environment
    NULL - (static) precreated empty environment (end, in env chain)
```

Spring 2020

Code Example

- Env.java
- EnvNode.java
- Bindings.java
- Binding.java
- Val.java