

Names (identifiers)

- Sir Walter Scott: author of Waverly
- "The king did not know Walter Scott was the author of Waverly"
 - we would be able to substitute "Walter Scott" with "the author of Waverly"
- "The king did not know the author of Waverly was the author of Waverly"
 - can't always do this because "Sir Walter Scott" has other meanings

C example:

```
int i = 27;
return i + 3; -> return 27 + 3

return &i + 1; -> return &27 + 1; (bad)

return sizeof(i); -> return sizeof(27) (bad)
```

- `i = 27`, longint 8, address of `i`
- name bound to:
 - value
 - type
 - address
 - alignment
- `alignof(i) ≠ alignof(27)`;
- binding: association between name + "value"
- `int i = 19`, established binding
 - `i = int(type), 19(val), 4(align), 0x79c00040(address)`

set of bindings (dictionary, namespace)

- names -> "values"
- the set of bindings can be determined
- explicitly; `int i; i -> int`
- implicitly; `fun i -> i + 1`

Binding Time

- when a name becomes bound to its value

- long int i = 27

i bound to	binding time
27	during execution when object is assigned to
long int	statically (compile time)
&i	block entry time (declaration time)

Names + Terminology

- declaration - gives enough info so people know what's going on
- `double difftime(time_t a, time_t b)`

Definition

- `double difftime(time_t a, time_t b) {return a-b}`

Namespaces

- nested declarations
- block structured scoped
- scope (of a name_):
 - set of locations in a program where the name is visible
- visible:
 - the name is written in program, has desired meaning

Information Hiding (for modularity)

- declare for each identifier whether its externally invisible
- C
 - static (private to this module)
 - extern (visible to all)
- Java

	in this class	another class in this package	subclass in some other package	other places where class is visible
public	X	X	X	X
protected	X	X	X	
(default)	X	X		

	in this class	another class in this package	subclass in some other package	other places where class is visible
private	X			

Ways to address bugs

- error (humans)
- fault (programs)
- failure (behaviors)
- compile-time checking (static checking)
- preconditions
- total definitions (Rust)
- fatal exists - abort()
- thread exception
- $\text{sqrt}(-1) \rightarrow \text{NaN}$