

## C Variables Can Be Global (Whole Program Scope)

But now we're going to write **C**!

In **C**,

- we use strings as datum names, so
- we could use a single scope.
- In other words, all data are available in all parts of the code.

**Good idea?**

**Absolutely not!**

## Picking Lots of Unique Names is Difficult

**Why not?**

**Managing global names is a nightmare.**

Imagine a program

- of 1,000,000 lines,
- with 20 programmers.

Can you pick names that are unique?

Did you remember that the program includes library code? What about the names there?

So ... **tiny gain, lots of pain.**

## Use File Scope in C, Not Global Scope

**Avoid defining global variables.**

C allows programmers to limit scope to

- a file,
- a function, or
- a compound statement.

For **file scope**,

- put the variable **outside of all functions**, and
- Write "**static**" in front of the declaration:

```
static int my_var; // usable in this file
```

## Local Variables Have Scope within a Function/Block

For **function/compound statement scope**,  
**declare** the variable **between braces**:

```
{ // sometimes called a "block"  
    int i, j;  
    // i and j can be accessed here  
}
```

Such variables are called **local variables**.