

Enumerations Start with 0 in C

In **C**, **enumerations**

- **start with 0**, but
- **can be overridden**.

For example, given

```
enum {FALSE, TRUE};
```

- **FALSE has value 0**, and
- **TRUE has value 1**.

Values are Numbered (and Re-Numbered) Automatically

We **count the number of values** by **adding an extra name** at the end:

```
enum {
    SPACE_EMPTY,
    SPACE_FULL,
    SPACE_BLOCK,
    NUM_SPACE_TYPES
};
```

If new names are added,

- **NUM_SPACE_TYPES grows automatically**
- along with any arrays based on it.

Enumerations Can Also Be Used for Bit Vectors

Enumerations can also be used **to name bits in a bit vector**:

```
enum {
    LEFT_WALL   = 1,
    RIGHT_WALL  = 2,
    UPPER_WALL   = 4,
    LOWER_WALL  = 8,
    HAS_EXIT    = 16
};
```

Notice how the default values can be overridden.