

▪ Speicherklassen (für Variablen)

- **static** (für alle Variablen möglich): Nur ein Zustand mit einmaliger Initialisierung, Zustand bleibt nach Funktionsaufrufen erhalten (bis Programmende)

```
void storage_class_static(int count) {  
    static int b = 1; ← Wird nur einmal initialisiert (rekursive Aufrufe, müssen entsprechend angepasst werden)  
    fprintf(stdout, "call %d #1: static int b == %d\n", count, b);  
    b = b * 2;  
  
    if(count < 2) {  
        storage_class_static(count + 1);  
    }  
  
    fprintf(stdout, "call %d #2: static int b == %d\n", count, b);  
    b = b * 2;  
}
```

static (bzw. eigentlich **extern**) ist **Standard-Speicherklasse von Top-Level-Variablen**

- **extern** (alle Variablen und Funktionen), **register** (lokale Variablen und Funktionsparameter): später...