

- Referenzzählung zur Erkennung von Speicherlecks
  - **Speicherreservierung** zu einem Datentypen zentralisieren (ideal: nur eine Funktion)
  - **Speicherfreigabe** zu einem Datentypen zentralisieren (ideal: nur eine Funktion)
  - **globale Zählvariable** je dynamisch allokiertem Datentyp
  - Zählvariable bei jeder Reservierung **erhöhen** und bei jeder Freigabe **vermindern**

```
unsigned int list_refs = 0;

List cons(List l, Element e) {
    List res = malloc( sizeof(*l) );
    list_refs++;
    ...
}

List deleteFirst(List l) {
    free(l);
    list_refs--;
    ...
}
```

```
unsigned int str_refs = 0;

String create(unsigned int len) {
    String s = malloc( sizeof(*s) );
    str_refs++;
    ...
}

String delete(String str) {
    free(str);
    str_refs--;
    ...
}
```

```
/* Hauptprogramm */
extern unsigned int list_refs;
extern unsigned int str_refs;
```