

- Implizit **Zeigerarithmetik** für den Zugriff auf Array-Elemente (Forts.)

```
int arr[4]
```

- **Ausnahme:** Als Parameter von `sizeof` bleibt ein Array ein Array  
=> `sizeof(arr)` liefert die echte Größe des Arrays, nicht die Größe eines Zeigers
- **Ausnahme:** Als Parameter von `&` wird `arr` nicht in einen Zeiger konvertiert  
(sondern bleibt ein Array) => `&(arr)` liefert Zeiger auf das Array
- **Kein Boundary Check:** Zugriff außerhalb der Arraygrenzen möglich

```
int arr[4] = { 0 };  
int      i = 0;  
for(i = -10 ; i < 10 ; i++) {  
    fprintf(stdout, "%d ", arr[i]);  
}
```