Kapitel 9: Vereinbarungen

Übersicht

- ✓ Eigenschaften von Vereinbarungen
  - ✓ Art der Vereinbarung
  - ✓ Gültigkeitsbereich
  - ✓ Bindung
  - ✓ Lebensdauer/Speicherklasse
- ✓ Speicherklassen-Attribute
- ➢ **Beispiele**
- Praxis: Header-Dateien
- Praxis: Funktionen und globale Objekte
- Rekursionen

Beispiel 1

```
void alpha ()
{
  int a;
  …
}
```

Eigenschaften von `a`:

- Datentyp: `int`
- Gültigkeit: lokal
- Sp.Kl.-Voreinstellung: `auto` (← lokales Objekt)
  - Speicherklasse: automatisch
  - Bindung: intern
  - Vereinbarung: Definition
  - Initialisierung: nein

Beispiel 2

```
void beta ()
{
  static int d;
  …
}
```

Eigenschaften von `d`:

- Datentyp: `int`
- Gültigkeit: lokal
- Sp.Kl.-Attribut: `static`
  - Speicherklasse: statisch
  - Bindung: intern
  - Vereinbarung: Definition
  - Initialisierung: ja (implizit mit Null)

Beispiel 3

```
static int d;
void gamma ()
{
  printf ("%d", d);
  …
}
```

Eigenschaften von `d`:

- Datentyp: `int`
- Gültigkeit: global
- Sp.Kl.-Attribut: `static`
  - Speicherklasse: statisch
  - Bindung: intern
  - Vereinbarung: Definition
  - Initialisierung: ja (implizit mit Null)

Beispiel 4

```
extern int c;
void delta ()
{
  printf ("%d", c);
  …
}
```

Eigenschaften von `c`:

- Datentyp: `int`
- Gültigkeit: global
- Sp.Kl.-Attribut: `extern`
  - Speicherklasse: statisch
  - Bindung: extern
  - Vereinbarung: Deklaration
  - Initialisierung: (nicht hier)

Beispiel 5

```
int b = 36;
void epsilon ()
{
  printf ("%d", b);
  …
}
```

Eigenschaften von `b`:

- Datentyp: `int`
- Gültigkeit: global
- Sp.Kl.-Voreinstellung: – (← global mit expliziter Initialisierung)
  - Speicherklasse: statisch
  - Bindung: extern
  - Vereinbarung: Definition
  - Initialisierung: ja (explizit)

Beispiel 6

```
int a;

void zeta ()
{
  printf ("%d", a);
  …
}
```

Eigenschaften von `a`:

- Datentyp: `int`
- Gültigkeit: global
- Sp.Kl.-Voreinstellung: – (← global ohne Initialisierung)
  - Speicherklasse: statisch
  - Bindung: extern
  - Vereinbarung: vorläufige Definition
  - Initialisierung: vorläufig mit Null

Beispiel 6

```
int a;

void zeta ()
{
  printf ("%d", a);
  …
}
```

Eigenschaften von `a`:

- Datentyp: `int`
- Gültigkeit: global
- Sp.Kl.-Voreinstellung: – (← global ohne Initialisierung)
  - Speicherklasse: statisch
  - Bindung: extern
  - Vereinbarung: vorläufige Def
  - Initialisierung: vorläufig mit N

Falls keine andere Definition vorhanden: **Definition** mit Initialisierung Null

Beispiele 1-6 im Skript …

Bspl.: außerhalb aller Blöcke (global):

	Speicherklasse	Bindung	Vereinbarung	Initialisierung
`int a;`	statisch	extern	vorläufige Def.	vorläufig 0
`int b = 36;`	statisch	extern	Definition	36
`extern int c;`	statisch	extern	Deklaration	nicht hier
`static int d;`	statisch	intern	Definition	0

Bspl.: innerhalb eines Blocks (lokal):

	Speicherklasse	Bindung	Vereinbarung	Initialisierung
`int a;`	automatisch	intern	Definition	nein
`int b = 36;`	automatisch	intern	Definition	36
`extern int c;`	statisch	extern	Deklaration	nicht hier
`static int d;`	statisch	intern	Definition	0

Beispiel

```
const int eins = 1;

void eta ()
{
  int e = eins;
  …
}
```

Eigenschaften von `eins`:

- Datentyp: `const int`
- Gültigkeit: global
- Sp.Kl.-Voreinstellung: – (← global mit expliziter Initialisierung)
  - Speicherklasse: statisch
  - Bindung: extern
  - Vereinbarung: Definition
  - Initialisierung: ja (explizit)

Beispiel

```
static const int
          zwei = 2;

void theta ()
{
  int z = zwei;
  …
}
```

Eigenschaften von `zwei`:

- Datentyp: `const int`
- Gültigkeit: global
- Sp.Kl.-Attribut: `static`
  - Speicherklasse: statisch
  - Bindung: intern
  - Vereinbarung: Definition
  - Initialisierung: ja (explizit)