

Prof. Markus Gross, Bruno Heidelberger, Richard Keiser, Nicky Kern, Edouard Lamboray, Christoph Niederberger, Tim Weyrich, Felix Eberhard, Manuel Graber, Nathalie Kellenberger, Marcel Kessler, Lior Wehrli

Uebung 4 - Musterlösung

1. Arrays

a) `char str = "a";`

Nicht erlaubt. "a" repräsentiert eine Zeichenkette (mit abschliessendem '`\0`' Zeichen) und kann somit keinem einzelnen `char` zugewiesen werden.

Mögliche Korrektur: `char str = 'a';`

b) `short array[4] = { 1, 2 };`

Erlaubt. Es müssen nicht alle Elemente eines Arrays initialisiert werden.

Die Variable `array` enthält 4 Werte des Typs `short`, wobei die letzten beiden Werte vom Compiler gleich Null gesetzt werden.

c) `char hello[] = "hello world";`

Erlaubt. Die Grösse des Arrays wird automatisch gross genug gewählt, um alle Werte der Initialisierung aufzunehmen.

Die Variable `hello` ist eine Zeichenkette mit Inhalt `hello world\0`. (Die Initialisierung terminiert automatisch die Zeichenkette mit einem '`\0`' Zeichen).

d) `float value[];`

Nicht erlaubt. Die Grösse eines Arrays muss spezifiziert werden, falls keine Initialisierung erfolgt. *Ausnahme:* Funktionsparameter.

Mögliche Korrektur: `float value[10];`

e) `unsigned int buffer[3] = { 13, 707, 0, 3000 };`

Nicht erlaubt. Die Initialisierung enthält mehr Elemente als das Array fassen kann.

Mögliche Korrektur: `unsigned int buffer[4] = { 13, 707, 0, 3000 };`

f) `double table[][] = { { 1.0, 0.0 }, { -1.0, 0.5 } };`

Nicht erlaubt. Nur die erste Dimensionsgrösse eines Arrays kann in Verbindung mit einer Initialisierung automatisch erzeugt werden.

Mögliche Korrektur: `double table[][2] = { { 1.0, 0.0 }, { -1.0, 0.5 } };`

2. Strings

Code: siehe Datei `u4a2_name.cpp`

3. Structs

Code: siehe Datei `u4a3_flaeche.cpp`