

Grundgerüst

Grundgerüst	Grundgerüst für einfache Programme
<p>Unser Grundgerüst besteht aus folgenden Komponenten:</p> <ul style="list-style-type: none">• Kommentare• Verfügbarmachung zusätzlicher Funktionalität (via <code>#include <...></code>)• main-Funktion <p>Beachte: Jedes ausführbare Programm braucht die main-Funktion.</p> <p>Beachte: Zusätzliche Funktionalität, wie beispielsweise Eingabe und Ausgabe, ist nicht in der "Grundfunktionalität" von C++ enthalten. Der Programmierer muss also mitteilen, wo diese zusätzliche "Funktionalität" definiert ist. Mittels <code>#include <...></code> kann er dem Compiler sagen, in welcher Befehlssammlung (genannt: Library) diese "Funktionalität" definiert ist. Ein- und Ausgabe sind beispielsweise in <code>iostream</code> definiert. Die Zeile <code>//#include "tests.h" // ...</code> wird für die Serien auf Codeboard verwendet, um dort die Programme zu testen und an den Übungsleiter abzugeben. (Hierzu werden die <code>//</code> am Zeilenanfang entfernt.)</p>	
<pre>// Informatik - Serie 13 - Aufgabe 4e // Programm: my_program.cpp // Autor: X. M. Mueller (Gruppe F) //#include "tests.h" // ... #include <iostream> int main () { // Your code here... return 0; }</pre>	

Datentypen

<code>int</code>	Datentyp für ganze Zahlen
<pre>int a = 3; int b = a + 4;</pre>	

Operatoren

*	Multiplikation von zwei <i>R</i> -Werten.
Präcedenz: 14 und Assoziativität: links	
<pre>int a = -3 * 4; int b = 2 * a; // Note: use l-value as r-value</pre>	

=	Zuweisungsoperator. Weist einem L-Wert einen neuen Wert zu.
Präcedenz: 4 und Assoziativität: rechts	
<pre>int a; a = 4; // value 4 a = 3; // value 3</pre>	

>>	(siehe: <code>std::cin</code> unter "Input/Output")
Präcedenz: 12 und Assoziativität: links	

<<	(siehe: <code>std::cout</code> unter "Input/Output")
Präcedenz: 12 und Assoziativität: links	

Input/Output

<code>std::cin >> ...</code>	Eingabe via Terminal (z.B. Tastatureingabe)
Erfordert: <code>#include<iostream></code>	
<pre>int a; std::cin >> a; // stores the user input into a</pre>	

<code>std::cout << ...</code>	Ausgabe in das Terminal
Erfordert: <code>#include<iostream></code>	
Das <code>\n</code> bewirkt einen Zeilenumbruch. Man kann stattdessen auch <code>std::endl</code> verwenden (siehe Beispiel unten).	
<pre>// tell the user to enter a number std::cout << "Enter height in metres: "; int h; std::cin >> h; std::cout << "Your input was: " << h << " m\n"; // line breaks std::cout << "This text is output..." << std::endl; std::cout << "This text is output...\n"; // does the same std::cout << "Twice the same was output...\n";</pre>	