Verbindungen zwischen LabVIEW und Visual Studio von Mikrosoft

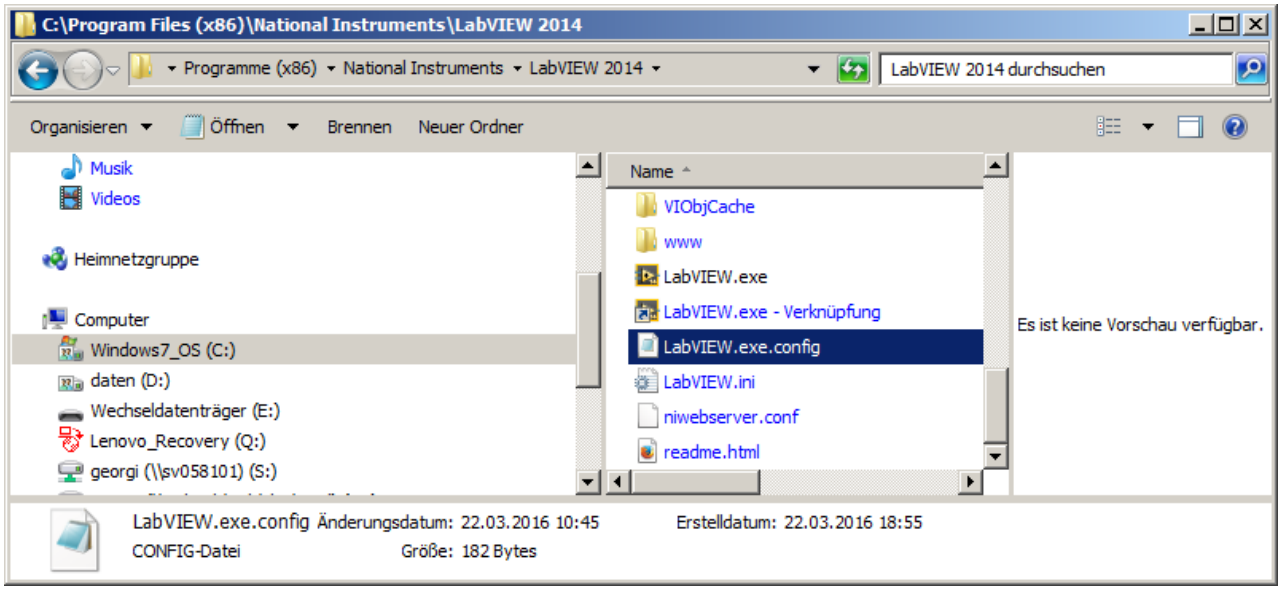
Ein Teil der Beispiele von Kapitel 9 befasst sich mit der Verbindung von LabVIEW-Programmen mit C-Programmen. Dabei tritt das Problem auf, dass die Entwicklungen bei National Instruments und Mikrosoft oft nicht hinreichend aufeinander abgestimmt sind. Daraus folgt, dass die in diesem Kapitel gegebenen Beispiele und Aufgaben, die unter LabVIEW 2010 zusammen mit Windows XP und Visual Basic 2008 nicht mehr laufen, wenn man auf LabVIEW 2014 und Windows 7 umstellt.

Fehler bei CSharp-Programmen

Hier ergibt sich das Problem, dass die unter 'Konnektivität' zu findenden '.NET' Funktionen, z.B. der Konstruktorknoten, nicht mehr laufen, wenn man das moderne '.NET Framework 4.0' verwendet. Die Frameworks 2.0, 3.0 und 3.5 verwenden CLR 2.0, nicht aber das Framework 4.0. Arbeitet man z.B. mit LabVIEW 2014, laufen die früher programmierten CSharp-VIs nicht mehr. Abhilfe bietet ein Programm LabVIEW.exe.config, das man mit einem Texteditor wie folgt schreibt:

```
<?xml version ="1.0" encoding="utf-8" ?>
<configuration>
<startup useLegacyV2RuntimeActivationPolicy="true">
<supportedRuntime version="v4.0.30319"/>
</startup>
</configuration>
```

Dieses Programm hat man in den Ordner zu bringen, in dem sich LabVIEW.exe befindet, also wie folgt:



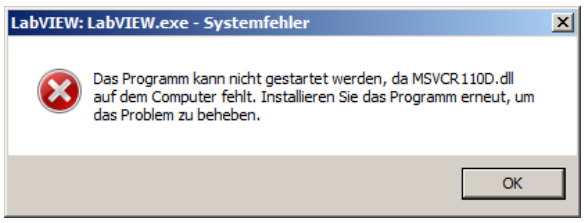
Danach Rechner herunterfahren und neu starten! Wir setzen - wie auch im nächsten Abschnitt - voraus, dass

- Betriebssystem Windows 7
- LabVIEW 2014
- Visual Studio 2015

installiert sind. Näheres dazu folgt anschließend.

Fehler bei Cplusplus-Programmen

Ruft man z.B. das Programm 'Reihe_Cplusplus.vi' im Ordner '0907-Reihe_Cplusplus' aus Auflage 5 des Lehrbuchs auf, erhält man folgenden Fehler:

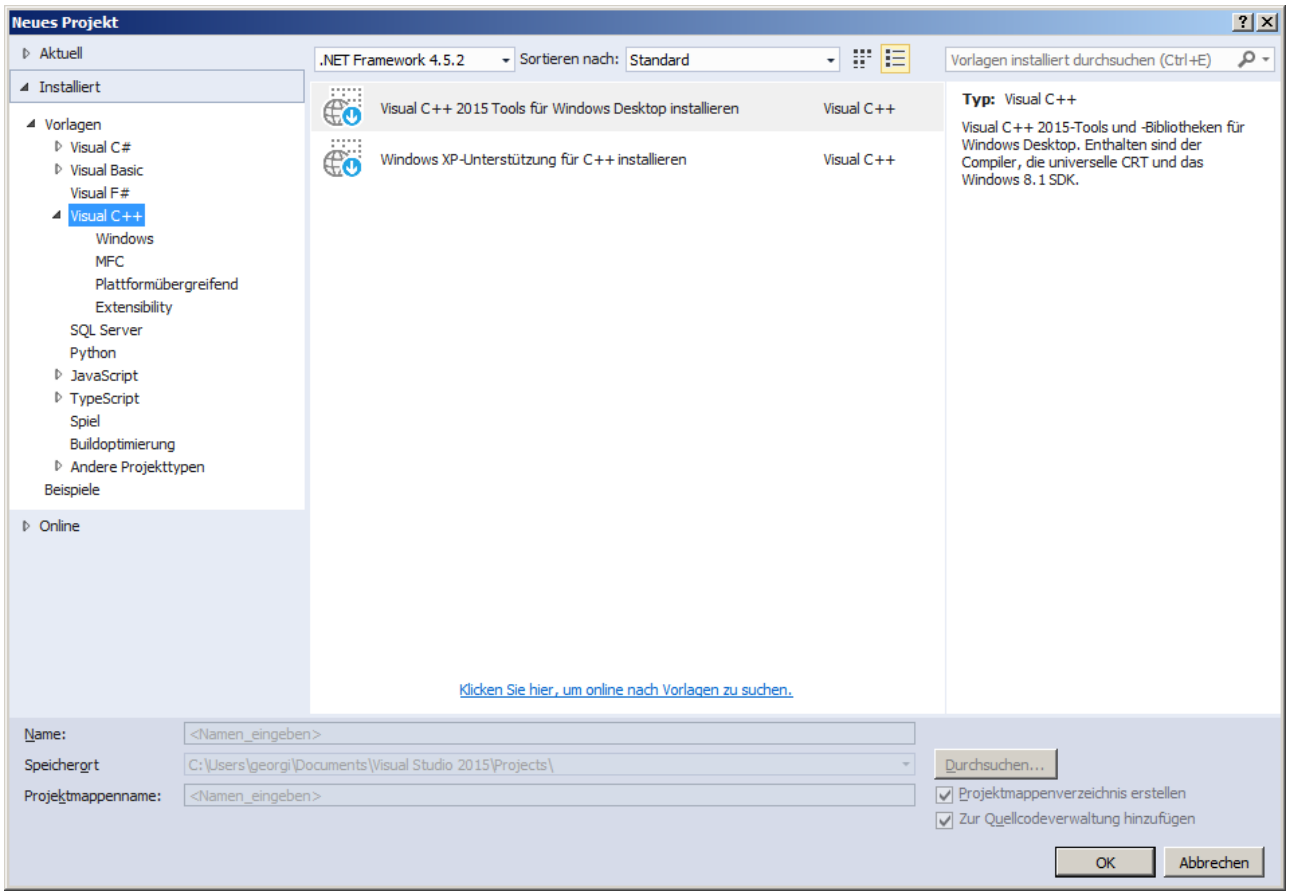


Im Internet findet man unter dem Stichwort 'MSVR110D.dll' verschiedene Rezepte. Hier verfolgen wir einen anderen Weg. Wir setzen voraus, dass wir mit dem

- Betriebssystem Windows 7,
- LabVIEW 2014 und
- Visual Studio 2015 arbeiten,

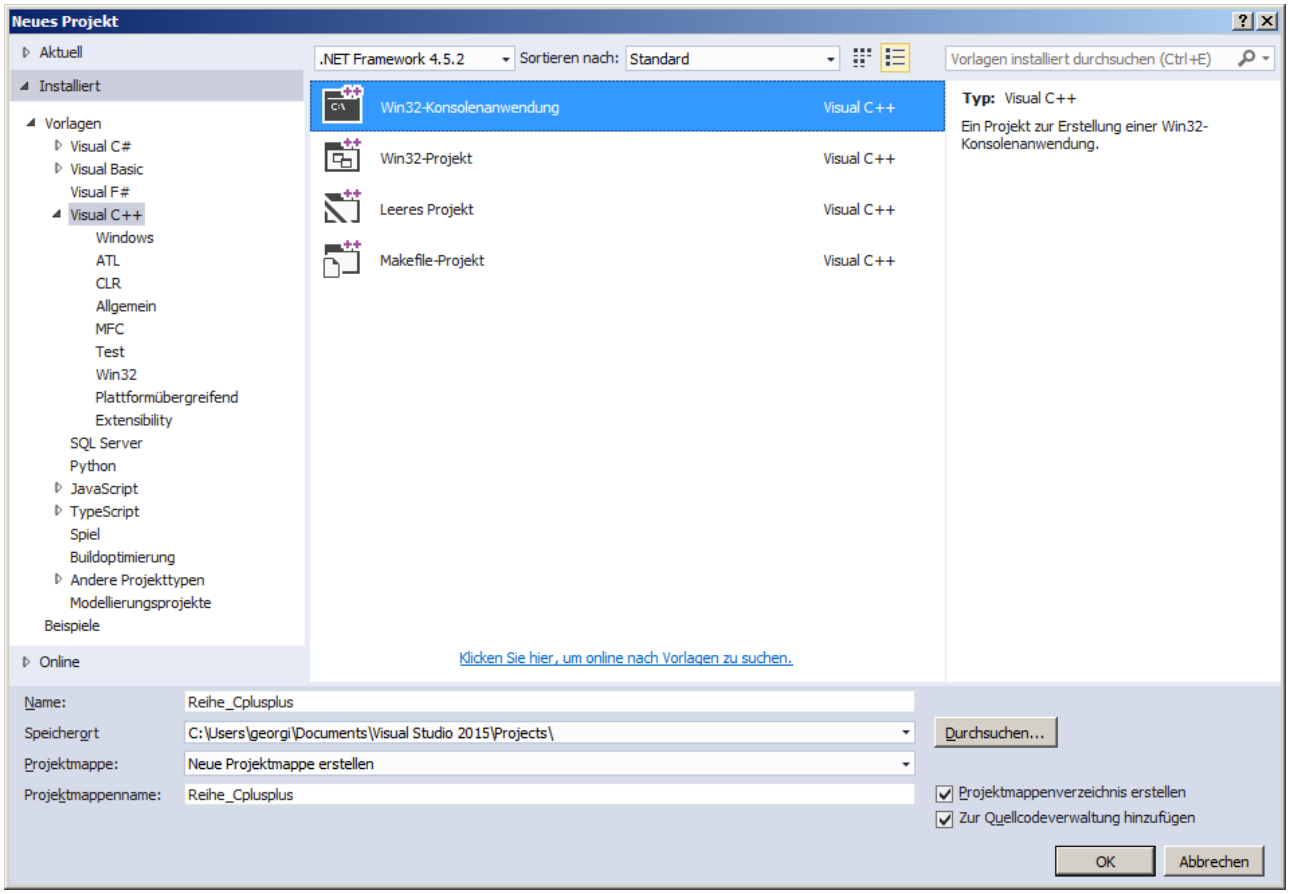
wobei wir zunächst noch Visual Studio 2015 installieren müssen.

Rufen wir dieses Programm unmittelbar nach der Installation von der Startleiste aus auf, bietet sich folgendes Bild:



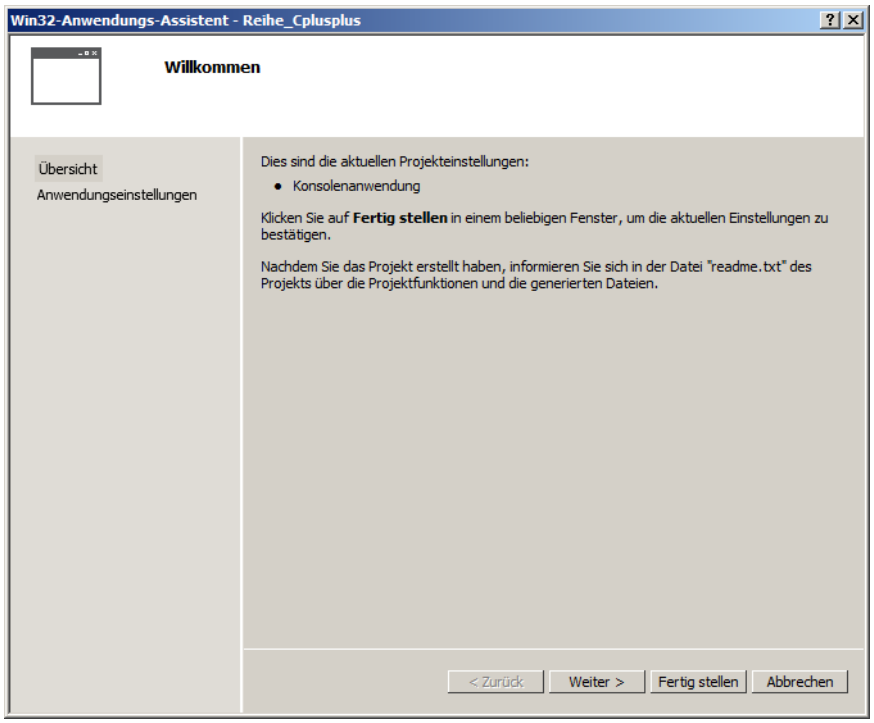
Das heißt, will man mit C++ arbeiten, benötigt man noch ein Zusatzpaket. Es genügt, das obere Paket 'Visual C++2015 Tools for Windows Desktop' zu installieren, was über 20 Minuten dauert.

Nun Visual Studio 2015 aufrufen und 'Neues Projekt' - 'Visual C++' - 'Win32-Projekt'. Dann erhält man folgendes Bild:

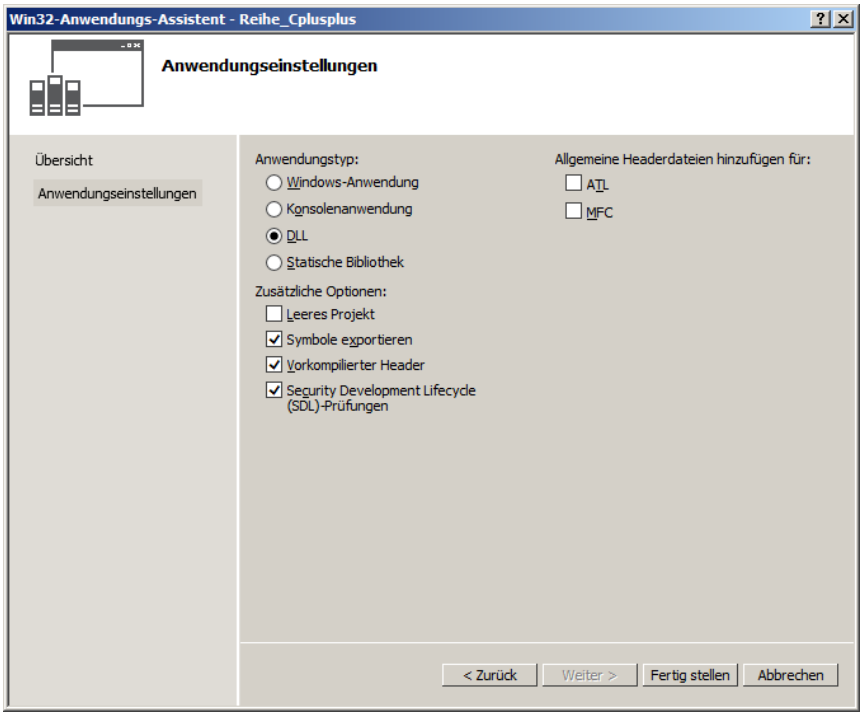


Für 'Name' und 'Solution name' trägt man ein: 'Reihe_Cplusplus'; für Location, was das System empfiehlt. Das wird natürlich je nach Rechneraufbau von 'C:\Users\georgi\...' abweichen. Danach 'OK'.

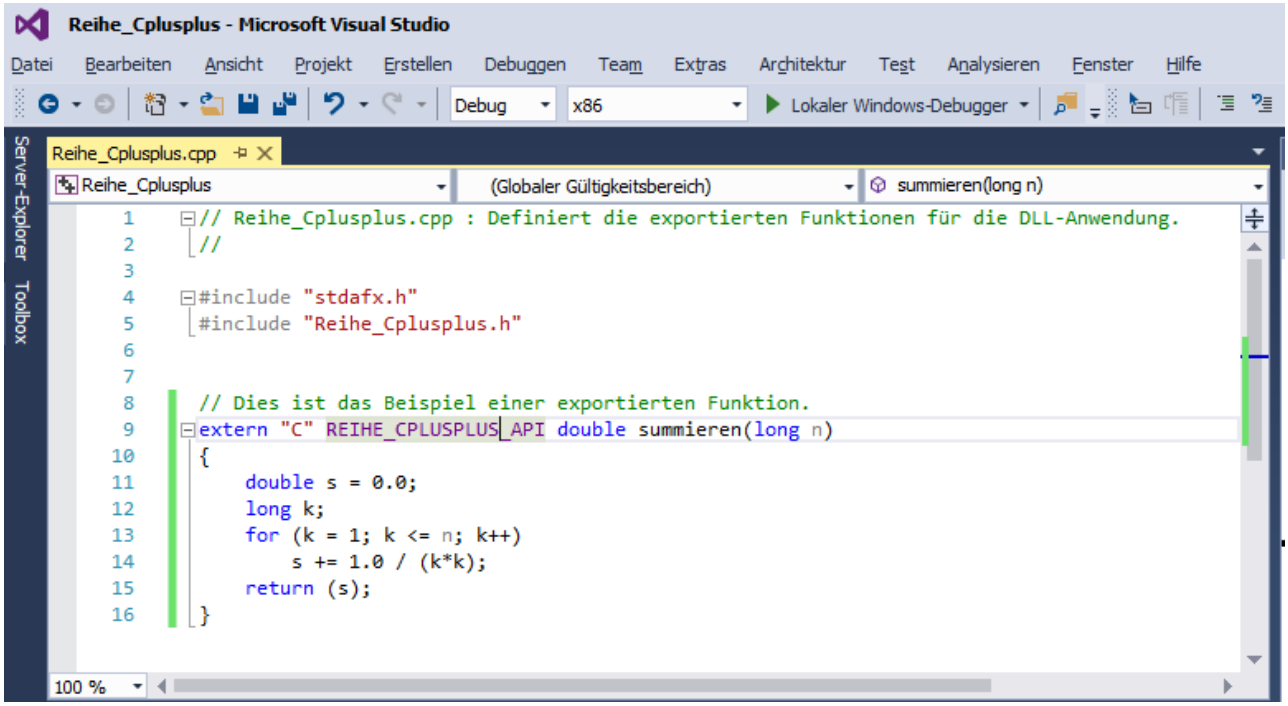
Im 'Win32-Anwendungs-Assistenten' (siehe nächstes Bild) wählt man 'Weiter'



und stellt gemäß folgendem Bild auf DLL' und 'Export Symbols' um:

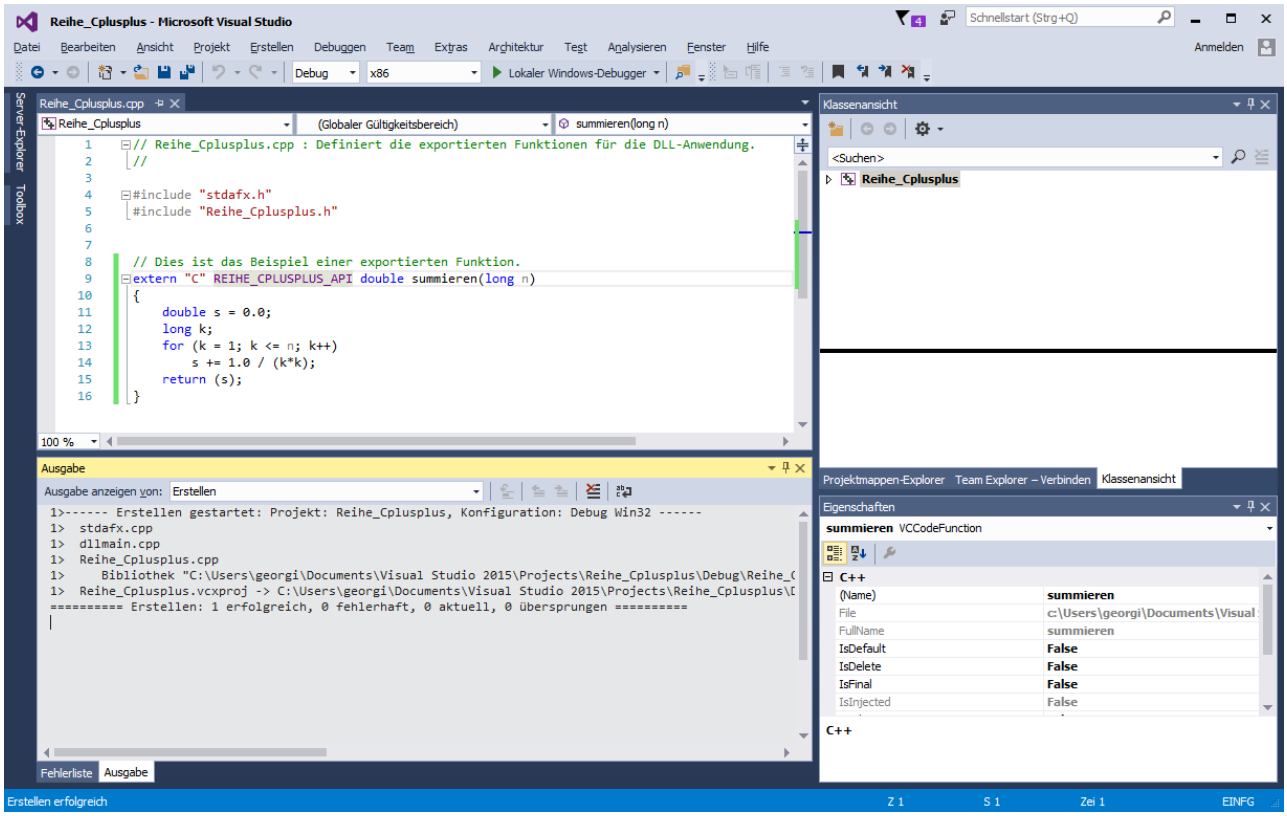


Nun schreibt man nach 'Fertigstellen' das C++-Programm wie im Lehrbuch beschrieben, wobei aber ein kleiner Unterschied im Aufruf zweckmäßig ist, weil das bei der Konfiguration der Call Library Function zu einfacher lesbaren Aufrufparametern führt: Man stellt vor den Funktionsaufruf ein ' extern "C" ', siehe dazu folgendes Bild:



Danach Erstellung aller erforderlichen Dateien mit 'Erstellen' - 'Reihe_Cplusplus erstellen'.

Das folgende Bild gibt eines Übersicht über den Prozess insgesamt:


Reihe_Cplusplus - Microsoft Visual Studio
Schnellstart (Strg+Q)
Datei Bearbeiten Ansicht Projekt Erstellen Debuggen Team Extras Architektur Test Analysieren Fenster Hilfe
Anmelden
Debug
x86
Lokaler Windows-Debugger
Server-Explorer
Toolbox
Reihe_Cplusplus.cpp
Reihe_Cplusplus
(Globaler Gültigkeitsbereich)
summieren(long n)
1 // Reihe_Cplusplus.cpp : Definiert die exportierten Funktionen für die DLL-Anwendung.
2 //
3
4 #include "stdafx.h"
5 #include "Reihe_Cplusplus.h"
6
7
8 // Dies ist das Beispiel einer exportierten Funktion.
9 extern "C" REIHE_CPLUSPLUS_API double summieren(long n)
10 {
11 double s = 0.0;
12 long k;
13 for (k = 1; k <= n; k++)
14 s += 1.0 / (k*k);
15 return (s);
16 }
100 %
Klassenansicht
<Suchen>
Reihe_Cplusplus
Projektmappen-Explorer
Team Explorer – Verbinden
Klassenansicht
Ausgabe
Ausgabe anzeigen von: Erstellen
1>------ Erstellen gestartet: Projekt: Reihe_Cplusplus, Konfiguration: Debug Win32 ------
1> stdafx.cpp
1> dllmain.cpp
1> Reihe_Cplusplus.cpp
1> Bibliothek "C:\Users\georgi\Documents\Visual Studio 2015\Projects\Reihe_Cplusplus\Debug\Reihe_C
1> Reihe_Cplusplus.vcxproj -> C:\Users\georgi\Documents\Visual Studio 2015\Projects\Reihe_Cplusplus\D
========== Erstellen: 1 erfolgreich, 0 fehlerhaft, 0 aktuell, 0 übersprungen ==========
Fehlerliste
Ausgabe
Eigenschaften
summieren VCCodeFunction
C++
(Name) summieren
File c:\Users\georgi\Documents\Visual
FullName summieren
IsDefault False
IsDelete False
IsFinal False
IsInjected False
C++
Erstellen erfolgreich
Z 1
S 1
Zei 1
EINFG