

Praktikumstermin Nr. 01, INF: Visual Studio Code, HalloWelt Programm

Ziel dieses Praktikumstermins

Sie machen sich mit der Entwicklungsumgebung *Visual Studio* vertraut. Ferner analysieren Sie einige Grundregeln der Programmiersprache C++.

Ziel hier: Vertraut machen mit der benutzten Entwicklungsumgebung, Analysieren einiger Grundregeln von C++.

Abgabemethodik bei diesem Praktikumstermin

Nachdem Sie alle Aufgaben für sich alleine erfolgreich durchgearbeitet haben, gehen Sie die Aufgaben noch einmal gemeinsam mit der Praktikumsbetreuung durch und beschreiben, was sie beim Durcharbeiten der Aufgaben für Ergebnisse beobachtet haben.

Ob Sie dabei die notwendigen Änderungen im Programm und das entsprechende Neucompilieren und Neustarten des Programms noch einmal wiederholen oder ob Sie das Programmverhalten aus ihrem Gedächtnis bzw. basierend auf etwaigen Notizen, die Sie vielleicht gemacht haben, erzählen entscheiden Sie nach Bedarf gemeinsam mit der Praktikumsaufsicht.

Visual Studio Code auf den FH Terminalrechnern

Sollten Sie das Praktikum an einem Terminalrechner der FH Aachen absolvieren, so brauchen Sie dort Visual Studio Code nicht mehr zu installieren, da es schon vorinstalliert ist.

Achtung: Es ist noch eine "Nicht-GIP-Version" von Visual Studio Code auf den Terminalrechnern installiert und deren Icon "drängt sich ggfs. in den Vordergrund". Starten Sie Visual Studio Code also nicht über ein ggfs. vorhandenes Desktop Icon oder über einen entsprechenden Startmenü-Eintrag, denn dann starten Sie wahrscheinlich eine Version ohne die GIP-spezifischen Anpassungen und Erweiterungen. Öffnen Sie stattdessen einen **Dateiexplorer** für das Verzeichnis `C:\vscodegip` und starten Sie die dort vorhandene Datei `code.exe` durch Doppelklick.

Aufgabe INF-01.01: (Bestandenen AGUM Test nachweisen)

AGUM = Arbeits-, Gesundheits- und Umweltschutzmanagement

Zur Nutzung der Praktikumsräume müssen Sie eine Sicherheitsunterweisung absolvieren. Dies umfasst Lesematerial und einen Abschlusstest. Zeigen Sie im

Praktikum ihren bestandenen AGUM Test vor. Absolvieren Sie den Test schon vor dem Praktikumstermin, nicht erst im Praktikum!

Link zum Test: https://www.ili.fh-aachen.de/goto_elearning_crs_267243.html

Aufgabe INF-01.02: (Projektordner und Arbeitsbereich anlegen)

Folgen Sie den Anweisungen der im GIP-INF Ilias Kurs verfügbaren Präsentation über die „Visual Studio Code – GIP Edition“, um Visual Studio Code zu installieren (falls eigener Rechner) und ein erstes Projekt anzulegen. Beachten Sie dabei:

- Nutzen Sie immer die aktuellste Version dieser Präsentation. Es ist durchaus möglich, dass es über das Semester hinweg immer mal wieder aktualisierte Versionen dieser Präsentation in Ilias geben wird.
- Benennen Sie das Projekt wie in der Präsentation angegeben mit `GIP-Prakt-01-Aufgabe-01`.
- Auf den Terminalrechnern unbedingt irgendwo unter Laufwerk H: speichern. Nur so stehen die dort gespeicherten Dateien auch automatisch auf anderen Terminalrechnern an gleicher Stelle zur Verfügung.
- Wählen Sie nach Anlegen des Projekts (mittels *Neues C++ Projekt erstellen* und den weiteren in der Präsentation beschriebenen Schritten zum Anlegen und Starten des ersten Projekts) im Menü *Datei* den Eintrag *Arbeitsbereich speichern unter ...* aus. Ändern Sie den vorgeschlagenen Dateinamen in `GIP-2023-2024-workspace` und wählen Sie als Verzeichnis für die Speicherung das Verzeichnis *eine Ebene höher als das vorgeschlagene Projektverzeichnis* aus, sprich das Verzeichnis, in dem sich das `GIP-Prakt-01-Aufgabe-01` Projektverzeichnis befindet. Speichern Sie die Arbeitsbereich-Konfiguration dort ab.

Aufgabe INF-01.03: (HaloWelt Programm)

Nutzen Sie das C++ `HaloWelt`-Programm, welches bei automatisch erstellt wird. Ergänzen Sie die Zeile `using namespace std;` und ändern Sie `std::cout` in `cout` sowie `std::endl` in `endl` gemäß dem `HaloWelt` Programm aus den Vorlesungsunterlagen.

Ändern Sie den Ausgabertext von `Hello, World!` in `Halo Welt!`

Compilieren und starten Sie das geänderte Programm aus Visual Studio Code heraus.

Testlauf:

Hallo Welt!

Drücken Sie eine beliebige Taste . . .

Aufgabe INF-01.04: (Relevanz `return 0` & `system("PAUSE")`)

Setzen Sie die Zeilen `return 0;` und `system("PAUSE");` in einen Blockkommentar. Was ist der Unterschied in der Ausführung verglichen mit Aufgabe INF-01.03 und welchen Rückgabewert des Programms beobachten Sie in Visual Studio?

Aufgabe INF-01.05: (Blockkommentare; Codeblöcke auskommentieren)

Ändern Sie den Blockkommentar aus Aufgabe INF-01.04 so, dass alle Zeilen im Rumpf des Hauptprogramms in den Blockkommentar eingeschlossen sind. Führen Sie das Programm aus. Was beobachten Sie? Was können wir also daraus folgern für ein C++ Hauptprogramm mit einem leeren Rumpf? Compiliert so ein Programm? Und wie sieht seine Ausführung aus?

Bemerkung: Dieses Vorgehen, einen Teil des C++ Codes über Kommentare zu „deaktivieren“, nennt man das *Auskommentieren* dieses (Teils des) Codes.

Aufgabe INF-01.06: (Relevanz von Zeilenumbrüchen)

In C++ werden Anweisungen mit einem Semikolon abgeschlossen. Verändern Sie den Quellcode so, dass Sie die C++ Anweisungen über mehrere Zeilen „strecken“, indem Sie z.B. bei `return 0;` das `return`, die `0` und das Semikolon in aufeinanderfolgende Zeilen schreiben. Compiliert und läuft das Programm noch?

Aufgabe INF-01.07: (Semikolon terminiert Anweisungen)

Nehmen Sie nun das Semikolon hinter einer der Anweisungen ganz weg. Compiliert und läuft das Programm noch?

Beobachten Sie die Syntaxprüfung, die schon in den Quelltext-Editor eingebaut ist: Suchen Sie rot unterschlingelte Teile des Quelltexts, fahren Sie mit der Maus darüber und warten Sie zwei Sekunden (Mauszeiger nicht bewegen). Was meldet das kleine Pop-Up für die Fehlerstelle.

Aufgabe INF-01.08: (Mehrere Anweisungen in einer Zeile)

Ergänzen Sie das Semikolon wieder und schreiben Sie mehrere Anweisungen in eine Textzeile des Quelltexts. Compiliert und läuft das Programm noch?

Aufgabe INF-01.09: (Anweisungsblöcke)

Setzen Sie jeden C++ Befehl (der sich ggf. über mehrere Zeilen des Quelltexts erstreckt) in eigene geschweifte Klammern {...}. Compiliert das Programm? Ändert sich etwas an der Programmausführung?

Aufgabe INF-01.10: (Einrückungen)

Einrückungen des C++ Programmcodes (z.B. im Rumpf des Hauptprogramms) dienen nur dazu, die Verständlichkeit des Quelltexts für den Programmierer zu erhöhen. Verifizieren Sie dies, indem Sie zuerst verschiedene Zeilen des Quelltexts verschieden stark einrücken und anschließend alle Einrückungen innerhalb des Quelltexts ganz wegnehmen.

Notlösung: FH Account funktioniert nicht & kein eigener Rechner

Für den Fall, dass ihr FH Account nicht funktionieren sollte (und Sie sich somit nicht an einem FH Terminalrechner einloggen können) und Sie auch nicht an ihrem eigenen Rechner arbeiten können, ist es für dieses Praktikum ausnahmsweise gestattet, eine Online C++ Programmierumgebung zu nutzen, z.B. unter <https://www.programiz.com/cpp-programming/online-compiler/>

Sie müssen uns dann aber zeigen können, dass Sie ihren FH Account frühzeitig getestet haben und seitdem im Dialog mit <https://hilfecenter.fh-aachen.de> sind, um ihr Account-Problem zu lösen. Ab dem zweiten Praktikum sollte diese Notlösung dann nicht mehr notwendig sein ...