

Walther- Rathenau- Gewerbeschule Freiburg	Aufgabe zu C-pointern SOL: selbst organisiertes Lernen Quatsch! SOL heisst: Schüler ohne Lehrer	Fach: ITS/FTI	Gruppe:
		25. November 2018	
		Name:	
		Klasse: E1FI2	Seite 1
		Punkte: /20	Note:

1 Pdf-Version dieses Aufgabenblatts

Die elektronische Version dieses Aufgabenblatts liegt hier:

```
http://dt.wara.de/sol/cPointer/aufgabe.pdf
```

2 Compilieren von C-Dateien unter Linux

Compiliert wird ein in C geschriebener Quelltext mit dem Kommando **gcc**. Haben wir z.B. die Datei `hello.c` geschrieben, kann man so in einem Terminal kompilieren:

```
gcc hello.c -o hallo
```

Dabei ist *gcc* ist der Compiler: GNU C Compiler, oder auch GNU Compiler Collection.

Natürlich muss man dem Compiler den Pfad zur Quelldatei, hier *hello.c* übergeben.

Optional kann man GCC noch sagen, wie die ausführbare Binärdatei heissen soll. Erreicht wird dies mit der Option **-o hallo**. Man ist bei der Wahl des Namens völlig frei. Es besteht keine Bindung zwischen dem Namen der Quelltextdatei und dem Namen der ausführbaren Datei.

Ausführen kann man das kompilierte Programm, indem man seinen Namen in der Shell (Terminal) eingibt:

```
./hallo
```

Der Compiler `gcc` ist ein Industriestandard und ist sehr komplex. Sein Handbuch hat knapp 1000 Seiten! Es gibt beliebig viele, beliebig komplizierte Optionen.

2.1 Fragen

- Fragen: wer ist **Richard Stallman**, was ist die **FSF** und was ist **GNU** (siehe <http://www.gnu.org>)?
- Was sind absolute Pfade, was sind relative Pfade, was ist das aktuelle Verzeichnis, wofür stehen die Zeichen `/`, `.` und `..` in einem Pfad?
- Wie sieht die Umgebungsvariable `PATH` aus und mit welchem Kommando kann ich sie mir anzeigen lassen? Welche Pfade enthält sie?
- Warum kann ich `hallo` nicht einfach so aufrufen, warum muss `./` davor?
- Wieviele Seiten hat das pdf-Handbuch des GCC Version 8.2.0 genau?
- Wer weitere Fragen zu `gcc` hat: `rtfm` = read the **fine** manual!

3 C-Pointer

Bitte folgendes durcharbeiten:

```
http://www.c-howto.de/tutorial/zeiger/  
http://www.c-howto.de/tutorial/zeiger/speicher/  
http://www.c-howto.de/tutorial/zeiger/beispiele/  
http://www.c-howto.de/tutorial/zeiger/uebung/
```

Und das hier, Kapitel 12.1 bis 12.3:

```
http://openbook.rheinwerk-verlag.de/c_von_a_bis_z/012_c_zeiger_001.htm
```