

Schritt für Schritt Anleitung: Ein Pascal Programm schreiben, kompilieren und ausführen

Giuseppe Accaputo

Informatik I (D-BAUG)

19. September, 2014

In diesem Dokument präsentiere ich euch eine kurze Schritt für Schritt Anleitung um euer erstes Pascal Programm zu schreiben, kompilieren und auszuführen.

Eine grobe Übersicht

1. In einem ersten Schritt schreiben wir unser Pascal Programm mit der Hilfe eines Texteditors. Das Resultat ist ein vom Menschen lesbarer Programmcode.
 - Das Ergebnis dieses Schritts ist eine Datei `HelloWorld.pas`, welche den Programmcode enthält
2. Nun möchten wir in einem nächsten Schritt den Programmcode so übersetzen, dass der Computer diesen auch versteht und für uns ausführen kann. Für diesen Schritt benötigen wir einen Compiler der für diese Übersetzung verantwortlich ist; in unserem Fall verwenden wir den Free Pascal Compiler `fpc`, da wir mit Pascal arbeiten.
 - Das Ergebnis dieses Schritts ist eine ausführbare Datei `HelloWorld`, welche vom Computer ausgeführt werden kann
3. In einem letzten Schritt führen wir unser Programm aus mit Hilfe der Befehlskonsole.
 - Das Ergebnis dieses Schritts ist die Ausführung des Programms und die eventuelle Ausgabe von Text auf der Konsole falls unser Programmcode `Write`, `Writeln` verwendet

Schritt für Schritt Anleitung: Vom Programmcode zum Programm

Schritt 1: Pascal Programm schreiben

Schritt 1.0: Texteditor auswählen

1. Um unser erstes Pascal Programm zu schreiben benötigen wir in einem ersten Schritt einen Texteditor. Unter folgender URL findet ihr eine Liste von Texteditoren für Windows, Mac OS X und Linux: http://de.wikipedia.org/wiki/Liste_von_Texteditoren.

Einige bekannte Beispiele:

- Windows: Notepad++
 - Mac OS X: TextMate
 - Linux: gedit
2. Nach der Installation könnt ihr den Texteditor starten und bereits eine neue, leere Datei erstellen. Dies erreicht man mit Hilfe des Menüs Datei → Neue ↪ Datei erstellen. Die meisten Editoren (Windows, Linux) verwenden die Tastenkombination CTRL + N um eine neue Datei zu erstellen; für Mac OS X könnt ihr CTRL mit der Apfeltaste ersetzen.

Schritt 1.1: Pascal Programmcode schreiben

1. Hat man den Texteditor mit einer leeren Datei vor sich offen, so kann man mit dem Schreiben des gewünschten Pascal Programms beginnen. In dieser Anleitung verwende ich folgendes Beispielprogramm:

```
PROGRAM HelloWorld;  
BEGIN  
    Writeln('Hello, world!');  
END.
```

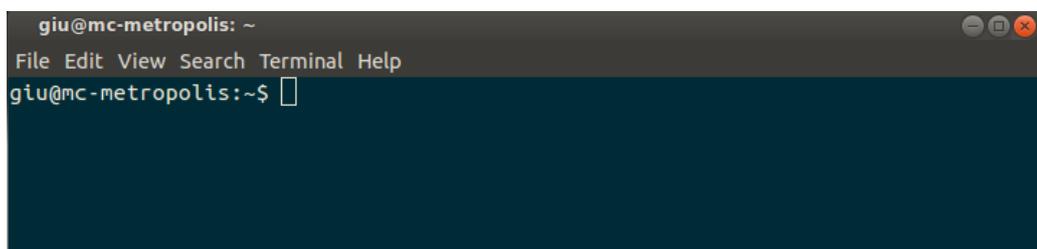
2. Speichert den Programmcode ab (tut dies alle paar Sekunden, ansonsten kann eure Arbeit beispielsweise bei einem Absturz verloren gehen).
 - (a) Bei der ersten Abspeicherung werdet ihr nach einem Dateinamen gefragt. Als Dateinamen könnt ihr den Namen verwenden, der neben dem PROGRAM ↪ -Schlüsselwort steht in eurem Programmcode kombiniert mit der Dateierweiterung .pas. Für unser Beispiel von oben solltet ihr die Datei also unter dem Namen HelloWorld.pas abspeichern.

- **Wichtig:** für die Übungsabgaben benötige ich nur die .pas Dateien von euch, da diese den Programmcode beinhalten.

Schritt 2: Pascal Programmcode kompilieren (für den Computer übersetzen)

In diesem Schritt wollen wir nun den geschriebenen Programmcode für den Computer übersetzen, sodass dieser das Programm in einem nächsten Schritt dann ausführen kann.

1. Startet die Konsole (Terminal)

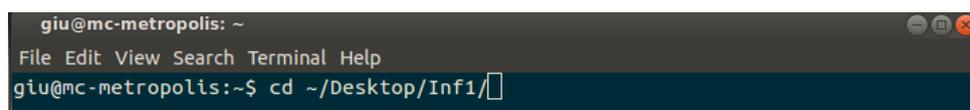


```
giu@mc-metropolis: ~  
File Edit View Search Terminal Help  
giu@mc-metropolis:~$
```

- Anleitung für Mac OS X: <http://www.blog-it-solutions.de/2011/mac-os-terminal-starten/>
- Anleitung für Windows 7/8: <http://praxistipps.chip.de/windows-konsole-oeffnen-1320>

2. Wechselt in den Ordner, in welchem ihr eure .pas Datei abgespeichert habt

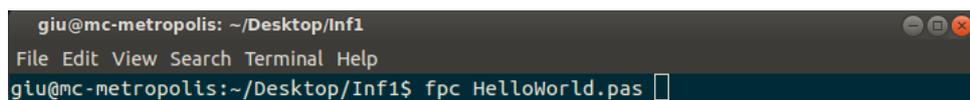
- Der `cd` (*change directory*) Befehl kann dazu verwendet werden. Gibt `cd C:/Inf1` in die Konsole ein (die .pas-Datei befindet sich dabei im Ordner C:/Inf1) und drückt die Taste ENTER



```
giu@mc-metropolis: ~  
File Edit View Search Terminal Help  
giu@mc-metropolis:~$ cd ~/Desktop/Inf1/
```

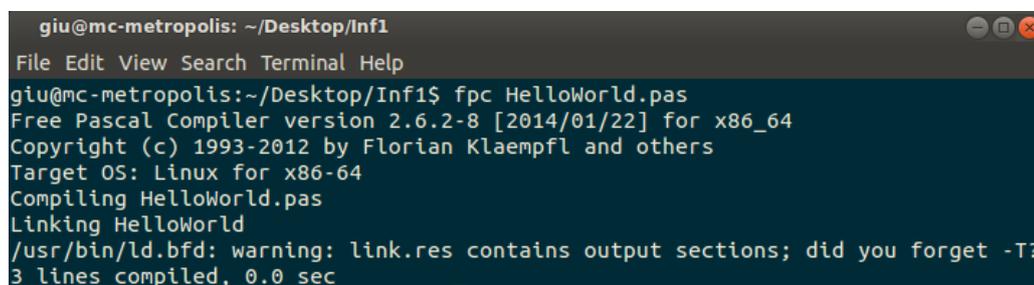
3. Nun rufen wir den Free Pascal Compiler `fpc` auf, um die .pas-Datei zu kompilieren (wir wollen die Datei für den Computer übersetzen)

- Für unser Beispiel könnt ihr in der Konsole folgendes eingeben und danach die ENTER-Taste drücken: `fpc HelloWorld.pas`



```
giu@mc-metropolis: ~/Desktop/Inf1  
File Edit View Search Terminal Help  
giu@mc-metropolis:~/Desktop/Inf1$ fpc HelloWorld.pas
```

4. Nach der Kompilierung des Programmcodes kann die Ausgabe auf der Konsole (Terminal) wie folgt aussehen:



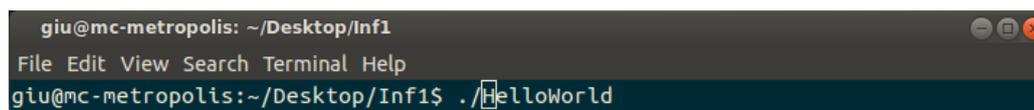
```
giu@mc-metropolis: ~/Desktop/Inf1
File Edit View Search Terminal Help
giu@mc-metropolis:~/Desktop/Inf1$ fpc HelloWorld.pas
Free Pascal Compiler version 2.6.2-8 [2014/01/22] for x86_64
Copyright (c) 1993-2012 by Florian Klaempfl and others
Target OS: Linux for x86-64
Compiling HelloWorld.pas
Linking HelloWorld
/usr/bin/ld.bfd: warning: link.res contains output sections; did you forget -T?
3 lines compiled, 0.0 sec
```

Der Programmcode wurde in diesem Fall erfolgreich kompiliert, da keine Fehler (z.B. *syntax errors*) ersichtlich sind in der Ausgabe. Auch könnt ihr sehen, dass die ausführbare Datei erstellt wurde. Dies ist in der Zeile `Linking HelloWorld` ersichtlich, wobei `HelloWorld` (in Windows `HelloWorld.exe`) die ausführbare Datei ist, welche vom Computer dann verwendet werden kann.

Schritt 3: Programm ausführen

Im vorherigen Schritt haben wir die ausführbare Datei `HelloWorld` generiert. Der Computer kann nun mit dieser Datei etwas anfangen, d.h. wenn wir sie ausführen wird der Computer das tun, was wir im Programmcode aus Schritt 1.2 definiert haben (in unserem Beispiel wird lediglich `Hello, world!` ausgegeben auf der Konsole)

1. Um die ausführbare Datei auf der Konsole (Terminal) auszuführen, müssen wir lediglich `./HelloWorld` in der Konsole eingeben (Windows: `HelloWorld.exe` statt `./HelloWorld` eingeben) und die ENTER-Taste drücken:



```
giu@mc-metropolis: ~/Desktop/Inf1
File Edit View Search Terminal Help
giu@mc-metropolis:~/Desktop/Inf1$ ./HelloWorld
```

2. Nach dem drücken der ENTER-Taste wird das Programm ausgeführt. Haben wir im Programmcode definiert, dass wir Text ausgeben möchten mittels `Write`, `↔ WriteLn`, so wird dieser Text in der Konsole angezeigt (in unserem Fall tun wir das; `Hello, world!` wird auf der Konsole ausgegeben):

```
giu@mc-metropolis: ~/Desktop/Inf1
File Edit View Search Terminal Help
giu@mc-metropolis:~/Desktop/Inf1$ ./HelloWorld
Hello , world !
```