

Eine kurze Anleitung zur Vorbereitung eines Apple-Rechners
im Sinne der Vorlesung
Informatik 251-0847-00 (D-MATH, D-PHYS),
Version HS08

Dorian Kind, Yves Brise

18. September 2008

Dieses Dokument bezieht sich auf die im Titel genannte Vorlesung, insbesondere auf die Programmierübungen, die fortlaufend auf http://www.ti.inf.ethz.ch/ew/courses/Info1_08/ veröffentlicht werden.

Da ein funktionierender Setup für das Bewältigen der Übungen vital ist, soll diese kleine Schrift Benutzern von MacOS X eine Hilfe sein, die benötigte Software mit vertretbarem Aufwand zu installieren. Es wird empfohlen, die Anleitung zunächst *ganz* durchzulesen.

Diese Version der Installationsanleitung bezieht sich auf Mac OSX 10.5 (Leopard). Wer Emacs auf einer früheren Version von OSX installieren will, sehe sich bitte eine frühere Version dieses Dokumentes an. Zu finden unter http://www.ti.inf.ethz.ch/ew/courses/Info1_07/

1 Installieren der XCode Tools (aka Developer Tools)

Der erste Schritt soll das Installieren der Entwicklungswerkzeuge sein. Auf der offiziellen Mac OSX “Leopard” Installations-DVD ist alles vorhanden, was das Herz begehrt. Wer keine hat, kann auf <http://ides.ethz.ch> für 20 Franken eine originale “Leopard” DVD erstehen, *oder* man kann bei der Apple Developers Connection (<http://developer.apple.com/products/>) den neuesten Installer gratis herunterladen (Anmeldung erforderlich).

Beim Doppelklicken auf `XcodeTools.mpkg` (siehe Abbildung 1) öffnet sich der Installer. Hier darf man sich erstmal etwas durchklicken und einen Lizenzvertrag akzeptieren, wonach dann das Installationsvolumen ausgewählt wird, und der interessantere Teil beginnt. Leider ist die Installation nicht gerade klein (2.6 GB). In Abbildung 2 sind die auszuwählenden Pakete angegeben. Unseres Wissens gibt es keine einfache Möglichkeit, die Grösse der Installation a priori zu verkleinern. Wer Platzprobleme hat, kann jedoch nach der Installation einige Dinge wieder löschen. Die Internet Seite http://www.askdaveytaylor.com/how_to_install_apple_developer_tools_cc_gcc_mac_os_x.html gibt dazu einige Tipps.

Das anschliessende Installieren ist eine ideale Gelegenheit, um sich einen Kaffee zu holen.

2 Ein Terminal/Konsole öffnen

Dies wird im nächsten Schritt gebraucht, ist aber ganz generell eine nette Sache.

Die zugrunde liegende Architektur von OSX ist ein Unix System. Somit hat man, abgesehen von der tollen graphischen Benutzeroberfläche, auch die Möglichkeit, das System mit Textbefehlen zu manipulieren. Dies kann ein sehr mächtiges Werkzeug sein. Zum Beispiel kann man den `g++` compiler dort direkt aufrufen.

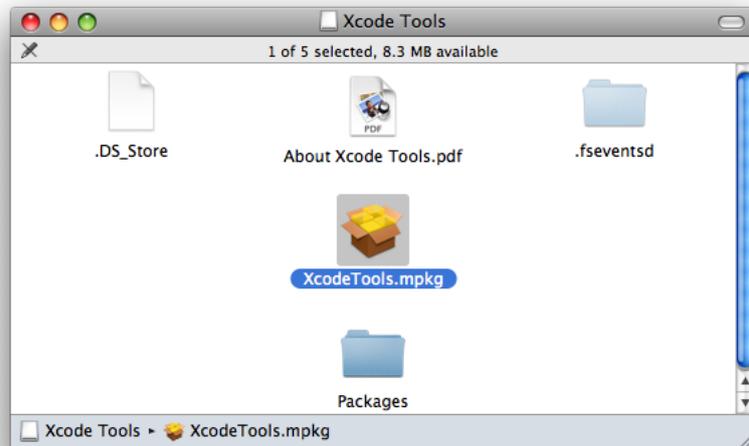


Abbildung 1: Die XcodeTools befinden sich auf der Installations-DVD oder können von der Apple Developers Connection heruntergeladen werden.

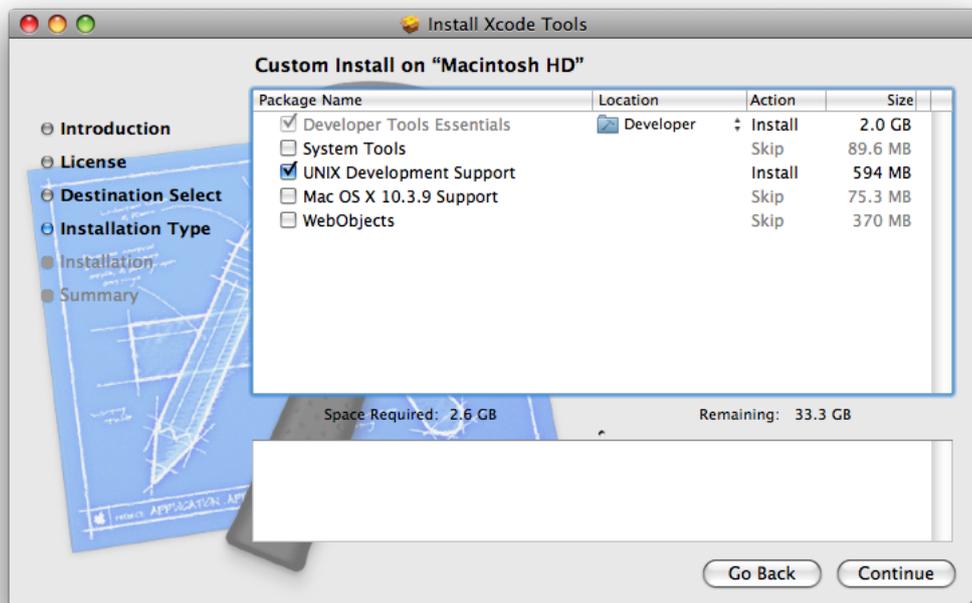


Abbildung 2: Die angegebenen Pakete von Xcode sollten installiert sein

Um ein Konsolen-Fenster zu öffnen, muss man die entsprechende Applikation unter `/Applications/Utilities/Terminal`¹⁾ doppelklicken. Nachdem man dieses Programm gestartet hat, kann man in einem Fenster Textbefehle eingeben und mit dem Drücken der Enter Taste zur Ausführung bringen. Wenn jemand generell mehr über die Konsole wissen will, dann empfehlen wir die Datei “Wichtige Unix Kommandos”, die auf der Vorlesungsseite zu finden ist oder Google: mac terminal.

3 Installieren von emacs

Hier hat sich seit der Einführung von Leopard zu Glück einiges vereinfacht. Wenn man Leopard installiert, wird emacs gleich mit installiert. Es ist allerdings nur eine Kommandozeilen Anwendung. Was man also noch machen muss, ist, einen sogenannten application wrapper einzurichten. Das geht in den folgenden einfachen Schritten.

- 1) Lade das emacs Quellpaket von ftp.gnu.org/gnu/emacs/ herunter. Die Datei, die benötigt wird, ist `emacs-22.1.tar.gz`.
- 2) Das heruntergeladene Archiv entpacken und die Datei `emacs-22.1/mac/Emacs.app` nach `/Applications/` verschieben. Die restlichen Archivdateien werden nicht mehr gebraucht und können gelöscht werden.
- 3) Als letztes muss man den application wrapper (die soeben verschobene Datei) mit der emacs Version des Systems (`/usr/bin/emacs`) verbinden. Das funktioniert, indem man eine Konsole öffnet und den Befehl

```
ln -s /usr/bin/emacs /Applications/Emacs.app/Contents/MacOS/emacs
```

eingibt. Dies geht davon aus, dass in Schritt 2) die Datei `Emacs.app` nach `/Applications/` verschoben wurde.

Das war’s. Ein Doppelklick auf `/Applications/Emacs.app` startet nun eine graphische Version von emacs.

Der Vollständigkeit halber sei angemerkt, dass es noch andere Installationen von emacs auf OSX gibt. Wir haben hier die unseres Wissens sauberste Lösung beschrieben, die am wenigsten Systemänderungen beinhaltet. Eine gute Übersicht über dieses Thema gibt <http://www.lshift.net/blog/2007/11/27/emacs-in-macos-x-105-leopard>.

4 Anpassen von emacs mittels des .emacs-files

Als letzter Schritt bleibt noch das Installieren des speziell für die Vorlesung erstellten `.emacs` Datei. Es ermöglicht Dinge wie das Drücken von F9 zur schnellen Kompilation und weitere nette Abkürzungen. Die Datei kann von der Vorlesungs-Homepage heruntergeladen werden, worauf sie typischerweise als `emacs.txt` gespeichert wird. Ein Verschieben der Datei in das Heimverzeichnis² und ein Umbenennen mittels `mv emacs.txt .emacs` in einem Konsolen Fenster, sollte die Sache berichtigen.

In Leopard kommt es zu Konflikten der F-Tasten zwischen der Konfigurations Datei `.emacs` und Exposé. Die Tasten F8, F9, F10 und F11 sind unter Umständen schon belegt und können nicht von emacs genutzt werden. Falls Exposé nicht gebraucht wird, kann es in den *Systemeinstellungen* unter *Dashboard und Exposé* ausgeschaltet werden. Alternativ kann man natürlich auch die Datei `.emacs` in einem Texteditor (z.B. emacs) ändern und seine gewünschte Tastaturbelegung einrichten. Dazu müssen ganz am Ende der Datei die Zeilen

¹bzw. `/Programme/Dienstprogramme/Terminal`

²Für den Benutzer `ybrise` wäre das `/Users/ybrise/`; im Finder wird das Heimverzeichnis durch ein stilisiertes Haus repräsentiert.

```

;; some keydefs
(global-set-key [f6] 'indent-whole-buffer)
(global-set-key [f7] 'previous-error)
(global-set-key [f8] 'next-error)
(global-set-key [f9] 'cxx-compile)
(global-set-key [f10] 'goto-line)

```

editiert werden. Es müssen einfach die entsprechenden f-Werte ersetzt werden (siehe Abbildung 3).

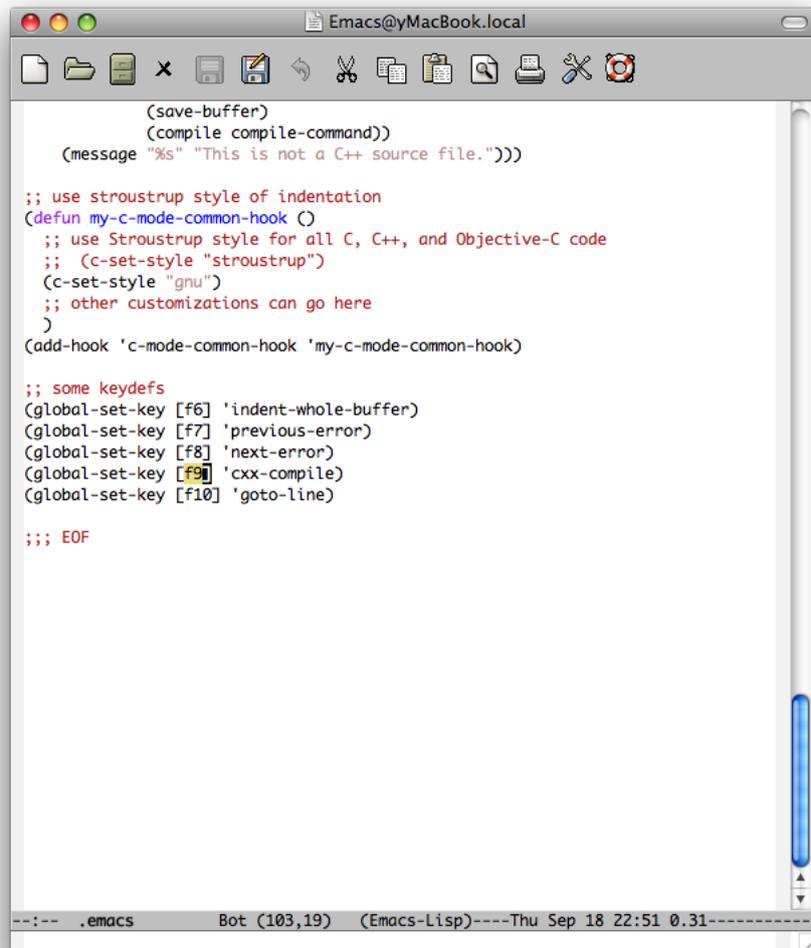


Abbildung 3: Die Markierung und der Cursor zeigen an, wo die Belegung für das Kompilieren von F9 auf etwas anderes gesetzt werden kann.

5 Epilog

Vielen Dank an Dorian Kind, der frühere Versionen dieses Dokumentes unterhalten hat.

Wir können nur hoffen, dass diese bescheidene Anleitung einigen Lesern nützlich ist. Um Verbesserungsvorschläge, Kritik und Erfahrungsberichte sind wir natürlich froh. So long!

Yves Brise (ybrise@inf.ethz.ch)