
Klausur

Prüfungsfach: Open-Source Software
Datum/Uhrzeit: 25. Juli 2006 / 10:30 Uhr
Raum: J201, J202
Prüfer: Dr. Hubert Högl
Dauer: 90 Minuten
Hilfsmittel: keine

Hinweise:

1. Dieses Klausurangabenblatt hat auch eine **Rückseite!** Bitte sofort überprüfen.
2. Schreiben Sie bitte nicht auf das Angabenblatt. Verwenden Sie für Ihre Antworten die separat ausgeteilten Bögen. Die Angaben dürfen Sie behalten.
3. Schreiben Sie nicht mit Bleistift.

Viel Glück!

Aufgabe 1 (8 Punkte)

- (a) Nennen Sie drei Konfigurationswerkzeuge, die dazu entworfen wurden, um Quelltext automatisch für verschiedene Plattformen zu kompilieren.
- (b) Sie haben ein Quelltextarchiv `src.tar.gz` und möchten es auf Ihrem Rechner unter Linux installieren. Was müssen Sie tun?
- (c) Sie haben ein Quelltextarchiv `cool-1.0.tar.gz` für Unix/Linux und möchten es unter Microsoft Windows kompilieren. Wie gehen Sie vor, so dass möglichst wenig Aufwand nötig ist?
- (d) Wo schauen Sie nach, wenn Sie wissen möchten welcher Kodierstil bei GNU Projekten empfohlen wird?
- (e) Welche Option verrät Ihnen die Version des `gcc`?
- (f) Welche Option des `gcc` extrahiert die *dependencies*?
- (g) Sie müssen schnell eine Man-Page schreiben. Wie gehen Sie vor?
- (h) Sie möchten schnell verschiedene Versionen einer lokale Datei verwalten. SVN und CVS sind Ihnen viel zu umfangreich. Welches Werkzeug wählen Sie? Bitte mit Anwendungsbeispiel.

Aufgabe 2 (10 Punkte)

Bei dieser Aufgabe geht es um die GNU „autotools“.

- (a) Wie heißen die zwei wichtigsten Input-Dateien bei der Konfiguration mit `automake` und `autoconf`? In welche Zieldatei werden diese durch das Anwenden von `automake` und `autoconf` abgebildet?

- (b) Wozu dient das optionale `config.h.in`? Wie wird es weiterverarbeitet?
- (c) Unter welchen Bedingungen kann man `./configure` direkt aufrufen und was passiert dabei?
- (d) Nennen Sie fünf gebräuchliche Targets des generierten Makefiles.
- (e) Was verstehen Sie unter den *Maintainerfiles*?

Aufgabe 3 (4 Punkte)

Erläutern Sie genau den folgenden Code:

```

1  static struct option long_options[] =
2  {
3      {"help",    no_argument, &help_flag, 1},
4      {"version", no_argument, 0, 'v'},
5      {"file",    required_argument, 0, 'f'},
6      {0, 0, 0, 0}
7  };
8  /* ... */
9  c = getopt_long (argc, argv, "abc:d:f:v",
10                 long_options, &option_index);

```

Aufgabe 4 (4 Punkte)

Erläutern Sie alle notwendigen Schritte um bei einem Projekt das Werkzeug *exuberant ctags* einzusetzen.

Aufgabe 5 (4 Punkte)

Die beiden Open-Source Programmiererinnen Anna und Maria haben beide das Quelltextarchiv `hui-2.0.tar.gz`. Anna macht mehrere Änderungen bei den Quellen und möchte dies Maria mitteilen, so dass auch sie über die aktuellen Quellen verfügt. Stellen Sie die Kommandos die Anna und Maria tippen nebeneinander in zwei Spalten gegenüber.

Aufgabe 6 (4 Punkte)

Ihr Bekannter interessiert sich für das „Subversion“ Versionskontrollsystem. Sie haben bereits ein Repository unter `https://meinsvn.com/` angelegt und zeigen Ihm nun die wichtigsten Kommandos `import`, `co`, `ci`, `add`, `rm`, `update`, `mv`, `info`. Dazu haben Sie ein Demo-Projekt unter `./demo/` angelegt. Schreiben Sie die Kommandos hin.

Wie macht man einen „Schnappschuss“ der aktuellen Version der Software?

Aufgabe 7 (6 Punkte)

Ein Demo-Programm besteht aus `main.c` und `util.c`. Das Makefile soll daraus verschiedene Varianten von ausführbaren Programmen generieren, die z.B. statisch oder dynamisch gelinkt werden.

- (a) Ergänzen Sie die fehlenden, mit XXX markierten Zeilen im folgenden Makefile.
- (b) Verwenden Sie nun das `libtool` zum Kompilieren der Bibliotheken.

```

1  %.o: %.c
2      XXX generiere .o aus .c. Soll sich zum Debuggen
3      XXX als auch fuer shared library eignen.
4
5  nolib: main.o util.o
6      gcc -g -o main main.o util.o
7
8  static: libutil.a main.o
9      gcc -o main main.o -L. -lutil
10
11 libutil.a: util.o
12     XXX generiere libutil.a aus util.o
13
14 dynamic: main.o libutil.so.1 solink
15     gcc -o main main.o -L. -lutil
16
17 libutil.so.1: util.o
18     XXX generiere libutil.so.1 aus util.o
19
20 solink:
21     ln -sf libutil.so.1 libutil.so

```

Aufgabe 8 (4 Punkte)

Wie würden Sie vorgehen, wenn Sie ein kleines bis mittleres Open-Source Projekt auf Ihrem Rechner (Server) verwalten müssten? Welche Software würden Sie installieren, damit Sie den heute üblichen Funktionsumfang abdecken? Beschreiben Sie ausführlich die angebotenen Dienste.

Aufgabe 9 (4 Punkte)

Beschreiben Sie die wesentlichen Eigenschaften der GNU *General Public License*.

Aufgabe 10 (4 Punkte)

Wer hat wann auf welche Weise den Begriff Open-Source Software geprägt?

Aufgabe 11 (2 Punkte)

Sie haben Ihr Programm unter die GPL gestellt und möchten nun Code dazufügen, der mit der folgenden freien Lizenz versehen ist. Dürfen Sie das? Bitte mit Begründung.

```

Copyright (c) <YEAR>, <OWNER>
All rights reserved.

```

```

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

```

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the <ORGANIZATION> nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Aufgabe 12 (2 Punkte)

Beschreiben Sie wie die *Creative-Commons* Lizenz funktioniert.