

Aufgabe 5: Bibliotheken (8 Punkte)

25.11.2009

Die Aufgabe ist einzeln zu bearbeiten! **Keine Zweiergruppen!**

In dieser Aufgabe sollen Sie Ihre Prozessliste aus Aufgabe 4 als statische und die Halde-Implementierung aus Aufgabe 3 sowie die Prozessliste als dynamische Bibliothek bauen und mit Ihrer clash-Implementierung binden. Die Abgabe erfolgt in Form eines Makefiles, das die im Folgenden beschriebene Funktionalität bieten soll:

a) Statische Bibliothek

Erstellen Sie ein Target `libsp.a`, welches eine statische Bibliothek aus der Objektdaten der `plist`-Implementierungen baut. Erstellen Sie außerdem ein Target, das die für die Bibliothek benötigte Objektdaten baut, und ein Target `clash-static`, das ein statisch gebundenes `clash`-Binary mit diesem Namen erzeugt. Versuchen Sie probierhalber auch, das `halde`-Modul in die statische Bibliothek mit aufzunehmen und das statische `clash`-Programm damit zu binden. Dokumentieren Sie die Probleme, die dabei auftreten, sowie Ihre Erklärung (**nm(1)**) für diese Probleme in der Datei **doc/halde.txt**.

b) Dynamische Bibliothek

Erstellen Sie nun ein Target `libsp.so`, das eine dynamische Bibliothek aus den Objektdaten der Prozessliste und der Halde baut. Vergessen Sie nicht, dass die Objektdaten für dynamisch gebundene Bibliotheken mit speziellen Compileroptionen übersetzt werden müssen. Sie müssen also u. U. für die Objektdaten andere Namen verwenden als in Teilaufgabe a) und entsprechend gesonderte Targets erstellen. Erstellen Sie außerdem ein Target `clash`, das ein dynamisch gebundenes `clash`-Binary erzeugt.

c) Standard Targets

Ihr Makefile soll auch die Standardtargets `all`, `clean` und `install` enthalten. Das Target `all` soll das erste Target sein und sowohl die statisch als auch die dynamisch gebundene Variante von `clash` und der Bibliothek erzeugen. Das Target `install` soll `clash` und `clash-static` in das Verzeichnis `../bin.i386` und die Bibliotheken in das Verzeichnis `../lib.i386` installieren. Die Verzeichnisse sollen hierbei erzeugt werden, falls sie noch nicht vorhanden sind.

d) Testen Sie die Ausführung

Testen Sie nun sowohl die statisch gebundene als auch die dynamisch gebundene Variante. Vergessen Sie nicht, vor der Ausführung der dynamisch gebundenen Variante die Umgebungsvariable `LD_LIBRARY_PATH` auf den Pfad ihres `lib.i386` Verzeichnisses zu setzen. Vergleichen Sie außerdem die Codegröße (**size(1)**) der beiden von Ihnen erstellten `clash`-Varianten.

Hinweise:

- Für die Abgabe wird nur die Datei **src/Makefile** erwartet. Das Makefile darf keine ins `make`-Programm eingebauten Regeln voraussetzen. Es ist nicht notwendig, die Quellcodes von `halde`, `plist` oder `clash` erneut abzugeben. Zur Übersetzung der `halde` ist die Angabe der Compileroption `-D_XOPEN_SOURCE=500` notwendig. Sie können diese in dieser Aufgabe auch zum Übersetzen der restlichen C-Dateien verwenden.
- Stellen Sie sicher, dass nur solche Dateien neu gebaut werden, deren Abhängigkeiten aktueller als das Target sind. Korrekte Target-Abhängigkeitsbeziehungen sind für das Erreichen der vollen Punktzahl notwendig. Mit Ausnahme der drei Targets `all`, `clean` und `install` soll ihr Makefile keine weiteren Pseudotargets enthalten.
- Mit Solaris 10 wird keine statische Version der C-Bibliothek mehr geliefert. Verwenden Sie daher zum Testen die Linux-Rechner im CIP-Pool.

Abgabe: bis spätestens Dienstag, 8.12.2008, 17:30 Uhr