

Aufgabe 5: Bibliotheken (4 Punkte)

20.11.2008

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

In dieser Aufgabe sollen Sie Ihre Prozessliste aus Aufgabe 3 sowie die halde-Implementierung aus Aufgabe 4 sowohl als statische als auch als dynamische Bibliothek bauen und mit Ihrer smash-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 den Objektdateien der `plist`- und `halde`-Implementierungen baut. Erstellen Sie außerdem die notwendigen Targets, um alle für die Bibliothek benötigten Objektdateien zu bauen, und ein Target `smash-static`, das ein statisch gebundenes `smash`-Binary mit dem Namen `smash-static` erzeugt.

b) Dynamische Bibliothek

Erstellen Sie nun ein Target `libsp.so`, das eine dynamische Bibliothek aus den Objektdateien der `plist` und der `halde` baut. Vergessen Sie nicht, dass die Objektdateien für dynamisch gebundene Bibliotheken mit speziellen Compileroptionen übersetzt werden müssen. Sie müssen also u. U. für die Objektdateien andere Namen verwenden als in Teilaufgabe a), und entsprechend unabhängige Targets erstellen. Erstellen Sie außerdem ein Target `smash`, das ein dynamisch gebundenes `smash`-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 `smash` und der Bibliothek erzeugen. Das Target `install` soll `smash` und `smash-static` in das Verzeichnis `../bin.i386` und die Bibliotheken in das Verzeichnis `../lib.i386` installieren.

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 der beiden von Ihnen erstellten `smash`-Varianten.

Hinweise:

Für die Abgabe wird nur die Datei **src/Makefile** erwartet. Es ist nicht notwendig, die Quellcodes von `halde`, `plist` oder `smash` erneut abzugeben. Zur Übersetzung der `halde` ist die zusätzliche Compileroption `-D_XOPEN_SOURCE=500` notwendig. Sie können diese in dieser Aufgabe auch zum Übersetzen Ihrer `smash` 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.

Abgabe: bis spätestens Donnerstag, 4.12.2008, 14:00 Uhr (gleiche Deadline wie Aufgabe 4)