



FACHHOCHSCHULE DARMSTADT

FACHBEREICH INFORMATIK

MIKROPROZESSORPRAKTIKUM 1

WS2005/06

Termin 6

Einfache Algorithmen & Datenstrukturen

Lernziele:

Unterprogrammtechnik, Adressierungsarten, bedingte Befehle

Aufgabenstellung:

Implementierung eines Sortieralgorithmus in ARM Assembler

Was ist Sortieren ?

Gegeben ist ein Feld $R[1..N]$ von N Zahlen, ordne die Elemente des Feldes so, daß gilt: $R[1] \leq R[2] \leq \dots \leq R[N]$

Sortieren ist ein recht intensiv erforschtes Gebiet der Informatik. Es existieren verschiedene Algorithmen: Bubble Sort, Heap Sort, Quick Sort. In diesem Versuch soll der Bubble Sort Algorithmus in ARM Assembler implementiert werden.

Algorithmus: Bubble Sort:

B1. [Initialisiere SCHRANKE] Setze $SCHRANKE := N$

B2. [Schleife über j] Führe B3 durch für $j=1,2,\dots,SCHRANKE-1$ und gehe dann zu Schritt B4.

B3. [Vergleich/Tausch] Falls $R[j] > R[j+1]$, tausche $R[j]$ mit $R[j+1]$.

B4. [Tausch ?] Ist $SCHRANKE = 0$, so terminiert der Algorithmus, ansonsten verringere $SCHRANKE$ um den Wert 1 und fahre fort mit Schritt B2.

Bemerkungen:

Dieser Algorithmus sortiert in aufsteigender Reihenfolge von R_j .

Bubble Sort ist einfach, aber langsam.

Beispiel: $N=6$

Schranke nachher	6 5	5 4	4 3	3 2	2 1	1 0
R[1]	43	34	34	34	34	34
R[2]	34	43	43	43	35	35
R[3]	64	64	48	35	43	43
R[4]	64	48	35	48	48	48
R[5]	48	35	64	64	64	64
R[6]	35	64	64	64	64	64

fett Bubble Sort am Beispiel 35

Hinweise:

Sie arbeiten in dem Verzeichnis `/home/studi/termin6`. Das Modul *bubble.S* steht dort als Programmgerüst zur Verfügung in Form der Datei *vorgabe_bubble.S*.

Diese Datei kopieren Sie nach *bubble.S* und ändern die Zugriffsrechte durch:

```
cp vorgabe_bubble.S bubble.S
```

```
chmod 755 bubble.S
```

Mit dem Aufruf *make* wird das Programm erzeugt. Hierzu steht eine Make-Datei mit dem Namen *makefile* zur Verfügung.

Wie kann man den Algorithmus optimieren?

Weiterführende Informationen:

Wichtige Dokumentation (im Verzeichnis `/auto/doku`):

ARM Befehlssatz Übersicht: QuickReferenz.pdf

ARM Befehlssatz: ADS_AssemblerGuide_B.pdf

ARM Programmiertechniken: ProgrammingTechniques.pdf

Vorbereitung und Teilnahmevoraussetzung:

Folgende Kenntnisse werden vorausgesetzt und am Anfang des Praktikums geprüft:

Grundstruktur einer Schleife in Assembler

Programmierung von if-else Strukturen in Assembler

Arithmetische Operationen in Assembler

Indirekte Adressierung und load/store Befehle

Statusregister des Prozessors

Bedingte Sprungbefehle

Programmentwurf des Moduls *bubble.S*. Der Entwurf wird am Anfang der Veranstaltung vom Betreuer überprüft.

Bemerkung:

Die Teilnehmer müssen im Praktikum vorbereitet sein. Der Betreuer des Praktikums behält sich vor, das Verständnis einzelner Teilnehmer im Rahmen eines kurzen Fachgespräches abzufragen. Eine weitere Teilnahme am Praktikum wird im Wiederholungsfall verweigert. Es besteht Anwesenheitspflicht.