Grundlagen der Informatik III

Wintersemester 2010/2011

Jan Philip Matuschek, Wolfgang Heenes, Patrik Schmittat

TECHNISCHE UNIVERSITÄT DARMSTADT

2. Praktikum

Ausgabe: 17.11.2010; Abgabe: 01.12.2010, 23:59 Uhr

Abgabe der Programme: Die Abgabe der Programme erfolgt über den SVN-Server `https://ics.ra.informatik.tu-darmstadt.de/svn/`. Die Beantwortung der Theoriefragen erfolgt in einem eigenen Dokument (PDF).

Aufgabe 1: Berechnen von x^n

Seien x und n Ganzzahlen, $n \geq 0$. Ihr Programm kann das Ergebnis modulo 2^{32} berechnen, das heißt Integer Overflows können unberücksichtigt bleiben.

Die Eingabewerte x und n können Sie fest im **.data**-Bereich festlegen oder sie von der Kommandozeile lesen (recherchieren Sie gegebenenfalls selbst, wie das geht).

a) Schreiben Sie ein IA32-Assemblerprogramm, das x^n iterativ berechnet und das Ergebnis ausgibt. Bei jeder Iteration soll das Zwischenergebnis mit x multipliziert werden. Das Assemblerprogramm soll als Unterprogramm realisiert werden. Die Übergabe der Parameter beim Aufruf erfolgt als call by value. Die Rückgabe des Ergebnisses erfolgt durch ein Register.

b) Schreiben Sie ein IA32-Assemblerprogramm, das x^n gemäß der folgenden Definition rekursiv berechnet und das Ergebnis ausgibt. Das Assemblerprogramm soll als Unterprogramm realisiert werden. Die Übergabe der Parameter beim Aufruf erfolgt als call by value. Die Rückgabe des Ergebnisses erfolgt durch ein Register.

$$x^n = \begin{cases} 1 & \text{falls } n = 0 \\ \left(x^{\frac{n}{2}}\right)^2 & \text{falls } n \text{ gerade} \\ \left(x^{\lfloor \frac{n}{2} \rfloor}\right)^2 \cdot x & \text{falls } n \text{ ungerade} \end{cases}$$

c) Skizzieren Sie den Stack Ihres rekursiven Programms zum Zeitpunkt der größten Rekursionstiefe, wobei Ihr Programm mit den Parametern $x = 45$ und $n = 22$ aufgerufen wird.

d) Welche Laufzeiten haben Ihre Programme aus den Aufgabenteilen a) und b)?

Bestimmen Sie mit Hilfe der Instruktion `rdtsc` die Anzahl der Ticks, die Ihre Programme für die Berechnung benötigen. Die Instruktion `rdtsc` schreibt die Anzahl der Ticks, die zwischen dem Start des Rechners und dem Aufruf der Instruktion vergangen sind, in das Register `eax`.[1]

Führen Sie die Messung für verschiedene Werte von n durch, die unterschiedliche Größenordnungen haben. Dokumentieren Sie die Messergebnisse in Form einer Tabelle und eines Diagramms (x-Achse: Problemgröße, y-Achse: Ticks).

Aufgabe 2: Berechnen der Ackermannfunktion

Die Ackermannfunktion ist eine extrem schnell wachsende, rekursiv definierte Funktion. Auch die Anzahl der zur Berechnung notwendigen rekursiven Aufrufe der Funktion wächst extrem schnell.

Schreiben Sie ein IA32-Assemblerprogramm, das die Funktionswerte der Ackermannfunktion gemäß der Definition von Péter berechnet:

[1] Die Tick-Anzahl ist eigentlich eine 64-Bit-Zahl. Die 32 höherwertigen Bits schreibt `rdtsc` in das Register `edx`. Zur Vereinfachung können Sie aber darauf verzichten, den Wert in `edx` auszuwerten.

$$A(m,\ n) = \begin{cases} n+1 & \text{falls } m = 0 \\ A(m-1,\ 1) & \text{falls } m > 0 \text{ und } n = 0 \\ A(m-1,\ A(m,\ n-1)) & \text{falls } m > 0 \text{ und } n > 0 \end{cases}$$

Die Übergabe der Parameter beim Aufruf erfolgt als call by value. Die Rückgabe des Ergebnisses erfolgt durch ein Register.

Aufgabe 3: Quicksort

Schreiben Sie ein IA32-Assemblerprogramm, das ein Datenfeld mit vorzeichenbehafteten 32-Bit-Zahlen sortiert und die sortierte Folge ausgibt. Implementieren Sie dazu den Sortieralgorithmus Quicksort, der hier in Pseudocode gegeben ist:

```
procedure QUICKSORT(A, l, r)
    if l < r then
        p ← PARTITION(A, l, r)
        QUICKSORT(A, l, p − 1)
        QUICKSORT(A, p + 1, r)
    end if
end procedure

procedure PARTITION(A, l, r)
    x ← A[r]
    i ← l
    for j ← l to r − 1 do
        if A[j] ≤ x then
            A[i] ↔ A[j]
            i ← i + 1
        end if
    end for
    A[i] ↔ A[r]
    return i
end procedure
```

Beim Aufruf des Unterprogramms wird die Basisadresse des Datenfeldes übergeben. Die weiteren Parameterübergaben sind nicht festgelegt. Nehmen Sie zum Testen folgende Zahlen:
86,19,59,63,79,65,45,27,61,31,94,85,42,92,21,81,63,18,56,25,12,70,57,81,40,60,93