

Praktikum Rechnerarchitektur - Informatik BSc - Blatt IV

Die Collatz Vermutung - Übergabe Parameter aus der Kommandozeile

Die bisher unbewiesene Vermutung besagt, dass eine Collatz Folge, startend mit einer natürlichen positiven Zahl immer in den Zyklus 4, 2, 1, 4, 2, 1, 4, 2, 1.... mündet.

Die Collatz Folge wird für streng positive Zahlen n wie folgt konstruiert:

$$n_i = \begin{cases} \frac{n_{i-1}}{2}, & \text{wenn } n_{i-1} \text{ gerade} \\ 3 * n_{i-1} + 1, & \text{wenn } n_{i-1} \text{ ungerade} \end{cases}$$

Erstellen Sie ein Programm in Assembly, welches als Input eine Zahl n_0 aus der Kommandozeile übernimmt, und die mit n_0 startend Collatz Folge ausgibt, bis die Zahl 1 zum ersten Mal erreicht wird.

Parameter aus der Kommandozeile werden über dem Stack übergeben:

- Die Anzahl der übergebenen Parameter ***argc (argument count)*** wird in der Stelle `esp+4` gespeichert. Das heißt, wir haben `[esp+4]` übergebene Parameter. Diese Parameter werden als Character-Strings übergeben. Diese Zahl ist mindestens 1, weil auch der Befehl, mit dem das Programm ausgeführt wurde, beinhaltet ist. Normalerweise ist das der Name des Programms. Die Anzahl der übergegebenen Parameter `argc` ist also immer um 1 größer als die Anzahl der vom Nutzer angegebenen Parameter.
- Die Adressen aller Argumenten (Strings) sind in einem Parametervektor ***argv (argument vector)*** gespeichert. Der Zeiger für das erste Element dieses Vektors, also der Zeiger zum ersten Argument (normalerweise den Programmnamen) ist in der Stelle `[esp + 8]` gespeichert. Der Zeiger zum zweiten Argument ist dann in der Stelle `[esp+8]+4` gespeichert.

Die Argumente sind als Strings gespeichert - soll eine Integerzahl abgebildet sein, dann kann dessen Wert über die C-Funktion `atoi` ermittelt werden: Angenommen der Register ***reg*** enthält den Zeiger zum String, also die Adresse des ersten Charakters eines Strings, welches eine Integerzahl abbildet, dann wird dieser Integer mit den folgenden Befehlen in ***eax (Rückgabewert!)*** gespeichert:

```
push dword[reg]
call atoi
add esp, 4
```

Wenn das übergebene String keine Ganzzahl darstellt, gibt `atoi` den Wert 0 zurück.

Es muss von Ihrem Programm kontrolliert werden, dass der Nutzer tatsächlich und exakt nur eine streng positive Zahl angegeben hat. Wenn nicht, dann soll das Programm mit einer entsprechenden Meldung abbrechen.

Als Hilfestellung steht Ihnen ein kommentiertes Programm, welches die vom Nutzer angegebenen Parameter nutzt, zur Verfügung (Moodle oder git_public_RAP).

Für das dritte Testat müssen Sie bis Ende KW49:

Als Team ausschließlich in den Praktikumszeiten Ihrer Gruppe:

- Ihr Programm mit Übernahme der Parameter aus der Kommandozeile und Ausgabe der entsprechenden Collatzfolge auf einem Laborrechner zeigen und Fragen dazu beantworten können.
- Ihr Programm assemblieren, linken und mit unterschiedlichen vorgegebenen Eingabewerten ausführen (das Programm wird auch auf Fehlverhalten des Nutzers geprüft, z.B. Nutzer gibt keine streng positive Zahl, kein Argument oder mehr als ein Argument).

Das Programm ins Moodle hochladen (oben in den Kommentaren schreiben Sie bitte mit wem Sie im Team zusammenarbeiten).

Es ist wichtig, dass alle Studierende das Programm separat hochladen, damit die Testatvergabe in Moodle bestätigt werden kann!