

DS - Übung 05

U 5.1)

- a) Beschreiben Sie, welche Besonderheit alle GNU-Assembler-Befehle aufweisen, die mit 'l' enden.

Geben Sie die **Hex-Darstellung** des Ergebnisses der beiden folgenden Befehlsfolgen in **r8** bzw. **r9** an. Geben Sie die einzelnen Rechenschritte an.

```
.section .data
bigNr:
    .quad 0x1234567890ABCDEF
```

- b) # (in .section .text)
 movq bigNr, %r8
 addl \$0700, %r8d # Was steht jetzt in %r8?
- c) # (in .section .text)
 movq bigNr, %r9
 movl \$700, %r9d # Was steht jetzt in %r9?
- d) Programmieren Sie die beiden Beispiele und kontrollieren Sie mit dem Debugger, ob Ihre Ergebnisse korrekt sind. Übersetzen Sie das Programm mit **as**.

U 5.2)

Gegeben ist **bigNr** wie in 5.1)

```
.section .data
bigNr:
    .quad 0x1234567890ABCDEF
```

- a) Welchen Wert hat **rax** nach den folgenden Befehlszeilen? Geben Sie die einzelnen Rechenschritte an.

```
movq bigNr, %rax
movswl %ax, %eax
cltq
```

- b) Belegen Sie das Register **r11** mit **bigNr**.
Schieben Sie den Inhalt von **r11** um 8 Bit nach rechts (**signed**) und anschließend wieder um 8 Bit nach links (**unsigned**).
Verwenden Sie im zweiten Fall das Register **%cl** für die Anzahl der zu verschiebenden Bits.
Geben Sie bei jedem Schritt die Belegung des Registers **r11** an (hier keine Rechenschritte notwendig).
- c) Programmieren Sie die Schritte a) und b) aus Aufgabe 5.2). Sie können dafür das gleiche Programm wie in 5.1) verwenden.