

Aachen, den 24.5.2019

Dr. rer. nat. Stefan Lankes
Steffen Vogel, M. Sc.

KGÜ 3

Funktionen

Inhalte: Funktionen, C-Calling-Convention, Parameter

Lernziele: Funktionen, Enter, Leave

Aufgabe 1: Der Funktionsaufruf

- a) Entwickeln Sie eine Assembler-Funktion `fib(i)` zur Berechnung von Fibonaccizahlen. Die Funktion gibt in den trivialen Fällen ($i \in 0, 1$) das entsprechende Ergebnis zurück und ruft sich sonst rekursiv auf. Testen Sie Ihre Assembler-Funktion mittels eines einfachen C-Programmes.
- b) Das Qt-Projekt *PiqRig* wird im Lernraum zur Vorlesung als Vorlage bereitgestellt. Mit dieser Anwendung können Sie Bilddateien öffnen und auf dem Bildschirm darstellen. Ein Menüpunkt erlaubt das Aufrufen von drei selbstdefinierten Funktionen, die das Bild verändern können. Die Funktionen haben alle die Form
- ```
extern "C" void func(void* bitmap, int width, int height);
```

Implementieren Sie die drei Assemblerfunktionen, die

- das Bild invertieren,
- eine Diagonale zeichnen oder
- das Bild in Schwarzweiß umwandeln.

#### Hinweis zur Invertierung:

Hierbei ist eine bitweise Invertierung gemeint.

#### Hinweis zur Umwandlung in Schwarzweiß:

Die Bitmap speichert die einzelnen Pixel im RGB-Format (s. <http://de.wikipedia.org/wiki/RGB-Konzept>). Sind alle 3 Werte gleich ( $R=G=B$ ), so erscheint das Pixel in einem Grauton. Aber welcher Grauwert (Helligkeitswert) ist für eine bestimmte RGB-Kombination adäquat? Dies kann mit Hilfe des YPbPr-Farbmodells (s. <http://de.wikipedia.org/wiki/YPbPr>) aus den RGB-Werten errechnet werden. Beachten Sie, dass wir keine Fließkomma-Operationen kennen. Wie lässt sich die Formel umstellen, so dass die Multiplikation mit Ganzzahlen erfolgen kann?

### Hinweis zum Qt-Projekt:

Die benötigten Bibliotheken können Sie unter Ubuntu mit folgenden Befehl installieren:

```
sudo apt-get install qt5-default
```

Im git-Repository zur Vorlesung ist eine Vorlage zum Erstellen des Programms als auch eine ausführliche Anleitung<sup>1</sup> zu finden. Führen Sie im Projektverzeichnis den Befehl `qmake` aus um ein Makefile zu generieren. Anschließend kann das Programm `PiqRig` mittels `make` erstellt werden.

---

<sup>1</sup><https://git.rwth-aachen.de/os/gi4/blob/master/kguebung/kguebung03/PiqRig64/README.md>