

Hardwarenahe Programmierung / Angewandte Informatik

Übungsaufgaben – 13. November 2017

Aufgabe 1: Datum-Bibliothek

Schreiben Sie eine Bibliothek `dates.c` (mit Header-Datei `dates.h`) zur Behandlung von Datumsangaben.

Diese soll enthalten:

- einen **struct**-Datentyp `date`, der eine Datumsangabe speichert,
- eine Funktion **void** `date_print (date *d)`, die ein Datum ausgibt,
- eine Funktion **int** `date_set (date *d, int day, int month, int year)`, die ein Datum auf einen gegebenen Tag setzt und zurückgibt, ob es sich um ein gültiges Datum handelt (0 = nein, 1 = ja),
- eine Funktion **void** `date_next (date *d)`, die ein Datum auf den nächsten Tag vorrückt.

Aufgabe 2: Fehlerhaftes Programm: Primzahlen

Das nebenstehende Primzahlprogramm (Datei: `aufgabe-2.c`) soll Zahlen ausgeben, die genau zwei Teiler haben, ist aber fehlerhaft.

Korrigieren Sie das Programm derart, daß ein Programm entsteht, welches alle Primzahlen kleiner 100 ausgibt.

```
#include <stdio.h>

int main (void)
{
    int n, i, divisors;
    for (n = 0; n < 100; n++)
        divisors = 0;
    for (i = 0; i < n; i++)
        if (n % i == 0)
            divisors++;
    if (divisors = 2)
        printf ("%d_ist_eine_Primzahl.\n", n);
    return 0;
}
```

Aufgabe 3: Fehlerhaftes Programm: Rotierender Würfel

Das auf der nächsten Seite abgedruckte OpenGL-Programm (Datei: `aufgabe-3.c`) soll einen sich gleichmäßig drehenden Würfel darstellen, ist jedoch fehlerhaft.

(Die Datei `opengl-magic.h` sowie die zugehörige C-Datei `opengl-magic.c` sind die aus der Vorlesung bekannten Beispiel-Dateien, die Sie zusammen mit diesen Übungsaufgaben herunterladen können.)

- (a) Warum sieht man lediglich ein leeres Fenster? Welchen Befehl muß man ergänzen, um diesen Fehler zu beheben?

Nach der Fehlerbehebung in Aufgabenteil (a) zeigt das Programm einen sich drehenden Würfel.

- (b) Erklären Sie das Drehverhalten des Würfels.
- (c) Welche Befehle muß man ergänzen, um eine gleichförmige Drehung zu erhalten?

```

#include <GL/gl.h>
#include <GL/glu.h>
#include <GL/glut.h>
#include "opengl-magic.h"

float t = 0.0;

void draw (void)
{
    glClear (GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);
    set_material_color (1.0, 0.7, 0.0);
    glRotatef (t, 0.5, 1.0, 0.0);
    glutSolidCube (0.5);
    glFlush ();
    glutSwapBuffers ();
}

void timer_handler (int value)
{
    t += 0.05;
    glutPostRedisplay ();
    glutTimerFunc (50, timer_handler, 0);
}

int main (int argc, char **argv)
{
    init_opengl (&argc, argv, "Cube");
    glutMainLoop ();
    glutTimerFunc (50, timer_handler, 0);
    return 0;
}

```