

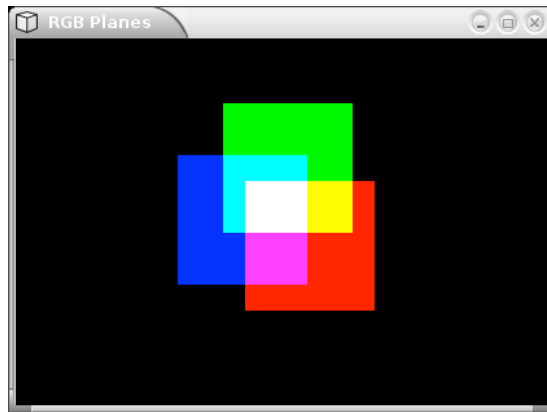
# Vorlesung Softwaretest und -debugging

Dr. Carsten Gnörlich

[cg@techfak.uni-bielefeld.de](mailto:cg@techfak.uni-bielefeld.de)

## Übungsblatt 8

21. Juni 2012



Im Übungsverzeichnis `/vol/lehre/softwaretest/uebungen/uebung08` finden Sie das Programm „rgb“, das Sie mit `make rgb` übersetzen können. Das Programm stellt drei Quadrate mit den Primärfarben dar, die Sie mit der Maus ziehen und so mischen können.

Das Programm ist nach dem Model/View/Controller-Ansatz implementiert. Seine Quellcode-Komponenten sind die folgenden:

- `rgb-model.c`: Die Datenhaltung des Programms.
- `rgb-viewer.c`: Darstellung der Quadrate mit der bekannten Grafikbibliothek.
- `rgb-controller.c`: Die Ereignisverarbeitung.
- `rgb-main.c`: `main()`-Routine und Zusammenbau der Komponenten.

Im Rahmen der Übungsaufgaben ist nur der Quellcode in `rgb-main.c` interessant. Die übrigen Programmteile können Sie als „black box“ betrachten.

## Aufgabe 1

Wir wollen eine (beliebige) Bedienungsfolge der graphischen Benutzeroberfläche aufzeichnen, um sie später als Testfall wieder abspielen zu können. Die Grundidee besteht darin, einen weiteren Observer in das System einzubauen, der die Mausbewegungen des Benutzers protokolliert und daraus ein passendes Programmkodestück generiert.

Für den Observer ist bereits ein passender Platzhalter in der Datei *rgb-main.c* mit dem Namen „recorder“ eingefügt. Sie müssen den Observer nur noch mit Programmcode füllen und den entsprechenden *RGBModelAttach()*-Aufruf einkommentieren.

Um die Mausbewegungen zu protokollieren, muß der Observer die folgenden Aktionen (Variable *action*) verarbeiten:

Aktion	Aufruf der Modellkomponente
OBS_MOUSE_MOVE	RGBModelMove(model, x, y)
OBS_MOUSE_PRESS	RGBModelPushButton(model, x, y)
OBS_MOUSE_RELEASE	RGBModelReleaseButton(model, x, y)

Aus der Variable *model->us\_elapsed* kann außerdem die Zeit entnommen werden, die seit der letzten Aktion vergangen ist.

Gestalten Sie den Observer *recorder* so, daß er eine per *#include* nutzbare Sequenz von Anweisungen für die Modellkomponente erzeugt. Beispiel:

...

```
RGBModelMove(model, 98, 63); usleep(180000);  
RGBModelPushButton(model, 98, 63); usleep(150000);  
RGBModelMove(model, 99, 63); usleep(1000);
```

...

Rufen Sie anschließend das Programm mit dem neuen Observer auf und generieren Sie eine kurze Protokolldatei:

```
> ./rgb >anim.h
```

## Aufgabe 2

Kommentieren Sie nun den Attach-Aufruf für den Observer *recorder* aus und füllen Sie den anderen Observer *player* mit Leben, um die eben aufgenommene Sequenz wieder abzuspielen.

Aufgrund von Einschränkungen der Graphik-Engine kann eine aufgezeichnete Sequenz erst dann abgespielt werden, wenn der Observer die Aktion *OBS\_START\_REPLAY* erhält. Auf diese Aktion müssen Sie mit einer if-Abfrage warten.

Wenn Sie die Benutzeraktionen wie oben beschrieben in der Datei *anim.h* protokolliert haben, können Sie den so generierten Quellcode per *#include* oder die Einfügen-Funktion Ihres Texteditors in den Observer einfügen.

Danach können Sie das Programm neu übersetzen. Es soll nach dem Aufruf automatisch die aufgezeichneten Aktionen durchführen.

Hinweis: Der Observer wird die Aktionen nicht zeitmaßstäblich ausführen, da einige Limitierungen im Motif-Toolkit dies verhindern. Eine zeitkorrekte Simulation der Nutzeroberfläche würde einiges mehr an Programmieraufwand benötigen.