



# Grundzüge der objektorientierten Programmierung

WS2001/2002 – Übungsblatt 8

Abgabetermin: 7. Januar 2002

## **Aufgabe 1.** *curses/ncurses, 5 Punkte*

Textaus- und -eingabe mit Cursorsteuerung erlaubt Ihnen die Terminal-Bibliothek `curses` (oder `ncurses` auf LINUX). Geben Sie bitte das folgende Programm als `curses.cc` ein, übersetzen Sie es dann mittels

```
CC -g -I/opt/local/include -o curses curses.cc -lcurses
```

oder

```
g++ -g -I/opt/local/include -o curses curses.cc -lcurses
```

oder auf LINUX-Systemen

```
g++ -g -I/opt/local/include -o curses curses.cc -lncurses
```

(ignorieren Sie eventuell auftretende Warnungen des Compilers). Lassen Sie es von einem `dtterm`-Fenster auf den Sun-Workstations beziehungsweise einem `xterm`-Fenster auf den LINUX-Systemen aus ablaufen.

```
#include <iostream>
#include <cstdlib>

// #include <curses.h>                für CC
// #include "/opt/local/gnu/include/curses.h"  für g++
//      (oder <curses.h> und g++ -g -I/opt/local/gnu/include ...

#ifdef __sparc
#ifdef __SUNPRO_CC
#include <curses.h>
#endif
#ifdef __GNUC__
#include "/opt/local/gnu/include/curses.h"
#endif
```

```

#else
#include <curses.h>
#endif

#include <csignal>

using namespace std;

// Übersetze mittels:
// CC -g -o 8_1 8_1.cc -lcurses (Sun)
// g++ -g -o 8_1 8_1.cc -lcurses
// bzw.
// g++ -g -o 8_1 8_1.cc -lcurses (LINUX)

extern "C" void finish(int sig);

int main(){

    int num = 0;

    /* initialize your non-curses data structures here */
    // ...

    (void) signal(SIGINT, finish);      /* arrange interrupts to terminate */

    (void) initscr();                  /* initialize the curses library */
    keypad(stdscr, TRUE);              /* enable keyboard mapping */
    (void) nonl();                     /* tell curses not to do NL->CR/NL on output */
    (void) cbreak();                   /* take input chars one at a time, no wait for \n */
    (void) echo();                     /* echo input - in color */

    if (has_colors())
    {
        start_color();

        /*
         * Simple color assignment, often all we need. Color pair 0 cannot
         * be redefined. This example uses the same value for the color
         * pair as for the foreground color, though of course that is not
         * necessary:
         */
        init_pair(1, COLOR_RED, COLOR_BLACK);
        init_pair(2, COLOR_GREEN, COLOR_BLACK);
        init_pair(3, COLOR_YELLOW, COLOR_BLACK);
        init_pair(4, COLOR_BLUE, COLOR_BLACK);
        init_pair(5, COLOR_CYAN, COLOR_BLACK);
        init_pair(6, COLOR_MAGENTA, COLOR_BLACK);
        init_pair(7, COLOR_WHITE, COLOR_BLACK);
    }
}

```

```

mvaddstr( 3, 10, (char *)"Dies ist ein Programm mit Menu:");
mvaddstr(10, 10, (char *)"Bitte geben sie einen der Buchstaben a,b oder c");
mvaddstr(11, 10, (char *)" (oder q zum Programmabbruch) ein:");

for (;;)
{
    move(11, 44);
// refresh();

    int c = getch();    /* refresh, accept single keystroke of input */
    attrset(COLOR_PAIR(num % 8));
    num++;

    /* process the command keystroke */

    mvaddstr(15,1, (char *)" ");
    clrtoeol();

    if ((c == 'a')||(c == 'b')||(c=='c')){
        mvaddstr(15, 20, (char *)"Sie drückten die Taste: ");
        addch(c);
    };
if (c == 'q') { beep(); finish(0);};
}

    finish(0);          /* we're done */
}

extern "C" void finish(int sig)
{
    endwin();

    /* do your non-curses wrapup here */
    // ...
    cout << "Programm beendet!" << endl;

    exit(0);
}

```

Versuchen Sie es Zeile für Zeile mit Hilfe der Kommentare, der Ergebnisse von `man curses`, `man mvaddstr`, ... beziehungsweise des [Webs](#) zu verstehen.

Schreiben Sie anschließend (in Aufsatzform) eine Beschreibung des Programms, die dessen Wirkungsweise Zeile für Zeile erklärt.

### **Aufgabe 2.** *try-catch, 5 Punkte*

Ändern Sie Ihr Programm aus Aufgabe 1 von Übungsblatt 5 so ab, daß der Erfolg/Mißerfolg von Speicheranforderungen per `new()` mittels geeigneter

try-catch Konstrukte sauber behandelt wird.

**Aufgabe 3.** *try-catch mit Exception-Hierarchie, 5 Punkte*

Ändern Sie Ihr Programm aus Aufgabe 4 von Übungsblatt 6 so ab, daß Exceptions `MathError` mit Unterexceptions `Overflow`, `ArgumentInvalid` genutzt werden. Benutzen Sie im testenden `main()`-Programm ein geeignetes try-catch Konstrukt.

**Aufgabe 4.** *Dateien, 5 Punkte*

Schreiben Sie ein C++-Programm, mit dessen Hilfe alle Kommentare aus einem aus einer Datei einzulesenden C++-Programmquelltext entfernt werden. Der modifizierte Programmquelltext soll in eine Datei ausgegeben werden, die eingelesene Datei soll unverändert bleiben. Benutzen Sie Exceptions bei den Dateioperationen.