



Betriebssysteme: Konzepte, Dienste,
Schnittstellen
(Betriebssysteme und betriebsystemnahe
Programmierung)

SS 2008 – Übungsblatt 3

Ausgabe: 7. Mai 2008

Abgabe: bis spätestens 21. Mai 2008

Aufgabe 1. *file/record locking*

Informieren Sie sich im Internet über den Begriff *file/record locking*.

Fassen Sie in eigenen Worten zusammen! Wie lautet der entsprechende deutsche Begriff?

Aufgabe 2. *time*

Informieren Sie sich mittels `man time` über die *real/elapsed time*, *USER cpu-time* und *SYSTEM cpu-time* bei der Ausführung eines Programms/Prozesses.

Fassen Sie in eigenen Worten zusammen. Führen Sie ein paar Zeitmessungen mit Hilfe von `time` durch und interpretieren Sie die Ergebnisse.

Aufgabe 3. *relocatable Code*

Was ist *relozierbarer Code*? Warum ist die Erzeugung *relozierbaren Codes* nötig?

Aufgabe 4. *reentrant Code*

Was ist reentrant Code? Warum benötigt man solchen Code? Durch welche Technologien kann er verwirklicht werden?

Aufgabe 5. *curses/ncurses*

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

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

auf Solaris beziehungsweise

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

auf Linux-Systemen. (Ignorieren Sie eventuell auftretende Warnungen des Compilers). Lassen Sie es von einem `xterm`-Fenster 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 -lncurses (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");

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.