Lösungen zur Vorlesung
GRUNDLAGEN DER INFORMATIK I
Studiengang Elektrotechnik
SS 03

AG Betriebssysteme FB3
Kirsten Berkenkötter

Übungszettel 7

1 Aufgabe 1

```
/*---------------------------------------------------------------------------*/
/* Programm zum Zaehlen von Buchstaben
 */
/*---------------------------------------------------------------------------*/

#include <stdio.h>

int main()
{
    FILE *datei;
    int i;
    int array[30];
    char d = ' ';
    unsigned char c;

    //Array mit 0 vorbelegen
    for(i = 0; i < 30; i++)
        array[i] = 0;

    //Datei oeffnen
    datei = fopen("probetext.txt", "r");

    //wenn geoeffnet wurde
    if(datei != NULL)
    {
        //Dateiende?
        while(d != EOF)
        {
            //naechstes Zeichen aus Datei holen
            d = fgetc(datei);

            //in unsigned char umwandeln
```

```
            //EOF ist -1, geht nur als char
            //erweiterter ASCII-Code geht nur als unsigned char
            c = (unsigned char) d;

            //Grossbuchstabe (von A=65 bis Z=90)?
            if((c >= 65) && (c <= 90))
                array[c-65]++;

            //Kleinbuchstabe (von a=97 bis z=122)?
            if((c >= 97) && (c <= 122))
                array[c-97]++;

            //ein AE?
            if((c == 228) || (c == 196))
                array[26]++;

            //ein OE?
            if((c == 246) || (c == 214))
                array[27]++;

            //ein UE?
            if((c == 252) || (c == 220))
                array[28]++;

            //ein eszett?
            if(c == 223)
                array[29]++;
        }

        //Anzahl der Buchstaben von a-z ausgeben
        for(i = 0; i < 26; i++)
            printf("%c: %d mal\n", i+65, array[i]);
        //Anzahl der Buchstaben fuer ae, oe, ue und eszett ausgeben
        printf("%c: %d mal\n", 196, array[26]);
        printf("%c: %d mal\n", 214, array[27]);
        printf("%c: %d mal\n", 220, array[28]);
        printf("%c: %d mal\n", 223, array[29]);

        //Datei schliessen
        fclose(datei);
    }
    //Fehler, wenn Datei nicht geoeffnet werden konnte
    else printf("Datei konnte nicht geoeffenet werden!\n");
}
```

2 Aufgabe 2

```
/*--------------------------------------------------------------------------*/
/* Programm zum Zaehlen von Sonderzeichen
 * Sonderzeichen sollen sein: alles ausser Buchstaben, Zahlen, Leerzeichen,
 * Tabs, Linefeeds und Carriage Returns
 */
/*--------------------------------------------------------------------------*/
#include <stdio.h>

//Werte fuer Leerzeichen, Carriage Return, Tab und Linefeed definieren
#define leer     32
#define cr       13
#define tab       9
#define lf       10

//Werte fuer ae, AE, oe, OE, ue, UE und eszett definieren
#define aek     228
#define uek     252
#define oek     246
#define aeg     196
#define ueg     220
#define oeg     214
#define eszett  223

int main()
{
    FILE *datei;
    int c = ' ';
    int zaehler = 0;

    int anzahl;

    //Datei oeffnen
    if((datei = fopen("probetext.txt", "r")) != NULL)
    {
        //am Ende der Datei?
        while( c != EOF)
        {
            //naechstes Zeichen holen
            c = fgetc(datei);

            //Zeichen ist EOF, Linefeed, Tab, Carriage Return oder Leerzeichen?
```

```
            //dann nicht weitermachen
            if((c != EOF) &&
               (c != lf) && (c != tab) && (c != cr) && (c != leer))
            {
                //Zeichen ist Umlaut oder Eszett?
                //dann nicht weitermachen
                if((c != aek) && (c != uek) && (c != oek) &&
                   (c != aeg) && (c != ueg) && (c != oeg) &&
                   (c != eszett))
                {
                    //Zeichen ist Buchstabe?
                    //dann nicht weitermachen
                    if(!(((c >= 65) && (c <= 90)) ||
                         ((c >= 97) && (c <= 122))))
                    {
                        //Zeichen ist Zahl?
                        //dann nicht weitermachen
                        if(!((c >= 48) && (c <= 57)))
                        {
                            //alles Uebrige ist ein Sonderzeichen
                            zaehler++;
                        }
                    }
                }
            }
        }
        //Ergebnis ausgeben
        printf("Sonderzeichen: %d mal\n", zaehler);

        //Datei schliessen
        fclose(datei);
    }
    else printf("Datei konnte nicht geoeffnet werden!\n");
}
```

3 Aufgabe 3

```
/*--------------------------------------------------------------------------*/
/* Programm zum Zaehlen von Woertern und Saetzen
 * Saetze enden mit Punkt, Rufzeichen oder Fragezeichen
 */
/*--------------------------------------------------------------------------*/
```

```
#include <stdio.h>

#define punkt        46
#define rufzeichen   33
#define fragezeichen 63

int main()
{
    FILE *datei;
    char wort[50];
    int ergebnis = 0;
    int zaehler = 0, zaehler2 = 0;
    char zeichen = ' ';

    //Datei oeffnen
    if((datei = fopen("probetext.txt", "r")) != NULL)
    {
        //am Ende der Datei?
        while( ergebnis != -1)
        {
            //naechstes wort holen
            ergebnis = fscanf(datei, "%s", wort);

            if(ergebnis != -1)
                zaehler++;
        }

        fseek(datei, 0, SEEK_SET);

        while( zeichen != EOF)
        {
            zeichen = fgetc(datei);

            if((zeichen == punkt) ||
               (zeichen == rufzeichen) ||
               (zeichen == fragezeichen))
                zaehler2++;
        }

        printf("Woerter: %d mal\n", zaehler);
        printf("Saetze: %d mal\n", zaehler2);
```

```
        //Datei schliessen
        fclose(datei);
    }
    else printf("Datei konnte nicht geoeffnet werden!\n");
}
```

4 Aufgabe 4

```
/*---------------------------------------------------------------------------*/
/* Programm zum Umdrehen von Wörtern
 * Jedes Wort einer Quelldatei soll umgedreht in eine Zieldatei geschrieben
 * werden. Pro Zeile sind 80 Zeichen erlaubt.
 */
/*---------------------------------------------------------------------------*/
#include <stdio.h>

#define punkt        46
#define rufzeichen   33
#define fragezeichen 63

int main()
{
    FILE *datei, *ziel;
    char wort[50];
    char umdrehen[50];
    int ergebnis = 0;
    int i;
    int zeichen = 0;

    //Datei oeffnen
    if((datei = fopen("probetext.txt", "r")) != NULL)
    {
        //Zieldatei oeffnen
        if((ziel = fopen("ziel.txt", "w")) != NULL)
        {
            //am Ende der Datei?
            while( ergebnis != -1)
            {
                //naechstes wort holen
                ergebnis = fscanf(datei, "%s", wort);

                //ein neues Wort eingelesen?
```

```
            if(ergebnis != -1)
            {
                //zeilenlaenge berechnen
                //+1 fuer Leerzeichen nach Wort
                zeichen += strlen(wort)+1;

                //evtl Zeilenumbruch einfuegen, max. 80 Zeichen
                if(zeichen >= 80)
                {
                    fprintf(ziel, "\n");
                    zeichen = strlen(wort)+1;
                }

                //Wort umdrehen
                for(i = 0; i < strlen(wort); i++)
                {
                    umdrehen[i] = wort[strlen(wort)-i-1];
                }
                umdrehen[i] = '\0';

                //umgedrehtes Wort in die Zieldatei schreiben
                fprintf(ziel, "%s ", umdrehen);
            }
        }
        //Zieldatei schliessen
        fclose(ziel);
    }

    //Datei schliessen
    fclose(datei);
  }
  else printf("Datei konnte nicht geoeffnet werden!\n");
}
```

5 Aufgabe 5

```
/*-------------------------------------------------------------------------*/
/* Programm zum Umwandlen von Gross- in Kleinbuchstaben
 * Jedes Wort einer Quelldatei soll in Kleinbuchstaben in eine Zieldatei
 * geschrieben werden. Nach jedem Satz (Punkt, Rufzeichen, Fragezeichen)
 * oder nach max. 80 Zeichen soll ein Zeilenumbruch erfolgen.
 */
```

```
/*---------------------------------------------------------------------------*/
#include <stdio.h>

#define punkt        46
#define rufzeichen   33
#define fragezeichen 63

int main()
{
    FILE *datei, *ziel;
    char wort[50];
    char neueswort[50];
    int ergebnis = 0;
    int i;
    int zeichen = 0;

    //Datei oeffnen
    if((datei = fopen("probetext.txt", "r")) != NULL)
    {
        //Zieldatei oeffnen
        if((ziel = fopen("ziel.txt", "w")) != NULL)
        {
            //am Ende der Datei?
            while( ergebnis != -1)
            {
                //naechstes wort holen
                ergebnis = fscanf(datei, "%s", wort);

                //ein neues Wort eingelesen?
                if(ergebnis != -1)
                {
                    //zeilenlaenge berechnen
                    //+1 wegen Leerzeichen nach Wort
                    zeichen += strlen(wort)+1;

                    //evtl Zeilenumbruch einfuegen, max. 80 Zeichen
                    if(zeichen > 80)
                    {
                        fprintf(ziel, "\n");
                        zeichen = strlen(wort)+1;
                    }
```

```
                    //Wort in Kleinschreibung aendern
                    for(i = 0; i < strlen(wort); i++)
                    {
                        if(wort[i] >= 65 && wort[i] <= 90)
                            neueswort[i] = wort[i]+(97-65);
                        else
                            neueswort[i] = wort[i];
                    }
                    neueswort[i] = '\0';

                    // Wort in die Zieldatei schreiben
                    fprintf(ziel, "%s ", neueswort);

                    if(((char *)strchr(neueswort, punkt) != NULL) ||
                       ((char *)strchr(neueswort, rufzeichen) != NULL)
||
                       ((char *)strchr(neueswort, fragezeichen) != NULL))
                    {
                        fprintf(ziel, "\n");
                        zeichen = 0;
                    }
                }
            }
            //Zieldatei schliessen
            fclose(ziel);
        }

        //Datei schliessen
        fclose(datei);
    }
    else printf("Datei konnte nicht geoeffnet werden!\n");
}
```