

```

/*****
 * Manipulationen mit Strings
 *
 * Uebung 3, Aufgabe 1 und 2
 *****/

#include<stdio.h>
#include<string.h>

#define MAXLAENGE 21 /* maximale Laenge des Strings */ 10

unsigned char wort[MAXLAENGE];

/* Berechnet die Laenge eines Strings */
short int string_laenge(void) string_laenge
{
    short int zeiger=0;

    while(wort[zeiger] != '\0') zeiger++; /* zeiger erhoehen bis Wortende erreicht */ 20

    return zeiger;
}

/* Invertiert einen String */
void string_invert(void) string_invert
{
    unsigned char buchstabe;
    short int zeiger=0, laenge; 30

    laenge=string_laenge()-1; /* Laenge-1, weil Indizierung bei 0 beginnt */
    /* 'laenge-zeiger' ist Zeiger auf Wortende */
    while(laenge-zeiger > zeiger) /* 'zeiger' ist Zeiger auf Wortanfang */
    { /* beide wandern aufeinander zu */
        buchstabe=wort[zeiger]; /* solange Wortende > Wortanfang */
        wort[zeiger]=wort[laenge-zeiger]; /* mache Zecks-Tausch */
        wort[laenge-zeiger]=buchstabe;
        zeiger++; 40
    }
}

/* Das Hauptprogramm */
void main(void) main
{
    printf("Gib den Sting ein: "); /* string einlesen */
    scanf("%20s", wort);

    /* Laenge berechnen */
    printf("Das Wort \"%s\" hat %d Buchstaben\n", wort, string_laenge()); 50
}
```

```

string_invert(); /* Buchstaben swappen */
printf("und rueckwaerts heisst es \"%s\".\n", wort);
}

/***** 60
 * Funktionen, die Gross-, Kleinbuchstaben vertauschen
 *
 * Uebung 3, Aufgabe 3
 *****/

#include<stdio.h> /* Funktionen laden */
#include<string.h>

#define MAXLAENGE 31 /* max. Laenge des Strings */ 70

unsigned char wort[MAXLAENGE]; /* glob. String */

/* wandelt alle Buchstaben in Grossbuchstaben um */
void in_grossb(void) in_grossb
{
    short int zeiger=0;

    while(wort[zeiger]!='\0') /* solange bis Stringende */ 80
    {
        if(wort[zeiger]>'Z') wort[zeiger]-=('a'-'A'); /* klein konvertieren */
        zeiger++;
    }
}

/* wandelt alle Buchstaben in Kleinbuchstaben um */
void in_kleimb(void) in_kleimb
{
    short int zeiger=0; 90

    while(wort[zeiger]!='\0') /* solange bis Stringende */
    {
        if(wort[zeiger]<'a') wort[zeiger]+=('a'-'A'); /* gross konvertieren */
        zeiger++;
    }
}

/* wandelt alle Buchstaben in Kleinbuchstaben um und umgekehrt */ 100
}
```

8616 Bytes  
13:35

muster.c  
1. Oct 1998

```
void buchst_tauschen(void)                                buchst_tauschen
{
    short int zeiger=0;

    while(wort[zeiger]!='\0')                               /* solange bis Stringende */
    {
        if(wort[zeiger]<'a') wort[zeiger++]+=('a'-'A'); /* 'A' > 'a' */
        else wort[zeiger++]-=('a'-'A'); /* 'a' > 'A' */
    }
}                                                         110

/* Hauptprogramm */
void main(void)                                           main
{
    printf("Gib das Wort ein: "); /* wort einlesen */
    scanf("%30s", wort);

    buchst_tauschen(); /* Gross-, Kleinschreiben */
    printf("Vertauscht: %s\n", wort); /* tauschen */ 120

    in_grossb(); /* 'a' > 'A' */
    printf("Gross geschrieben: %s\n", wort);

    in_kleimb(); /* 'A' > 'a' */
    printf("Klein geschrieben: %s\n", wort);
}

130
/*****
 * Vergleiche
 *
 * Übung 3, Aufgabe 4
 *****/

#include<stdio.h> /* Funktionen laden */
#include<string.h>

#define MAXLAENGE 31 140

unsigned char wort[MAXLAENGE];

/* sucht nach einem buchstaben */
int finde_buchstabe(char buchstabe) finde_buchstabe
{
    unsigned short int zeiger=0;

    while(wort[zeiger]!='\0') /* solange ende Wortende nicht erreicht */ 150
```

muster.c  
1. Oct 1998

8616 Bytes  
13:35

```
{
    if (wort[zeiger]==buchstabe) return 1; /* vergleiche Buchstaben */
    zeiger++;
}
return 0;
}

/* Hauptprogramm */
void main(void) main
{
    char buchst1, buchst2;
    int gefunden1, gefunden2;

    printf("Gib das zu ueberpruefende Wort ein: "); /* lese Daten ein */
    scanf("%s", wort);
    printf("Gib einen Buchstaben ein: ");
    fflush(stdin);
    scanf("%c", &buchst1);
    printf("Gib noch einen Buchstaben ein: "); 170
    fflush(stdin);
    scanf("%c", &buchst2);

    gefunden1=finde_buchstabe(buchst1); /* hole Suchergebnisse */
    gefunden2=finde_buchstabe(buchst2);

    /* !gefunden heisst: Negation von gefunden */

    if (gefunden1 && gefunden2)
        printf("Beide Buchstaben wurden gefunden.\n"); 180
    if ((gefunden1 && !gefunden2) || (!gefunden1 && gefunden2))
        printf("Einer der beiden Buchstaben wurde gefunden.\n");
    if (!gefunden1 && !gefunden2)
        printf("Keiner der beiden Buchstaben wurde gefunden.\n");
}

/*****
 * strings mit einer festen Zahl ascii-maessig verschieben
 *
 * Übung 3, Aufgabe 5a
 *****/

#include<stdio.h>
#include<string.h>

#define MAXSTRING 31

void main(void) main
{
    200
```

```
unsigned char wort[MAXSTRING];           /* Deklaration */
short int zahl, laenge;

printf("\nGib das Wort ein: ");           /* Eingabe */
scanf("%30s", wort);
printf("\nGib die Verschiebung ein: ");
scanf("%hd", &zahl);

laenge = strlen((char)wort);              /* Laenge des Wortes */

while (laenge--)                          /* solange bis 0.ter Buchstabe bearbeitet */
{
    wort[laenge] += zahl;
    if (wort[laenge] > 'Z')
        wort[laenge]=(wort[laenge]%('Z'-'A')+1)+'A'; /* Ueberlauf abfangen */
}

printf("\nDas neue Wort heisst: %s\n", wort);
}

/*****
 * Programm zum Dekodieren der kodierten Worte *
 *
 *      *
 *  Uebung 3, Aufgabe 5b
 *
 *****/

#include<stdio.h>
#include<string.h>

#define MAXSTRING 31

void main(void)
{
    unsigned char wort[MAXSTRING];        /* Deklaration */
    short int zahl, laenge;

    printf("\nGib das Wort ein: ");        /* Eingabe */
    scanf("%30s", wort);
    printf("\nGib die Verschiebung ein: ");
    scanf("%hd", &zahl);

    laenge = strlen((char)wort);          /* Laenge des Wortes */

    while (laenge--)                      /* solange bis 0.ter Buchstabe bearbeitet */
    {
        wort[laenge] -= zahl;
        if ((wort[laenge] < 'A') || (wort[laenge] > 'Z'))
            wort[laenge]=(wort[laenge]%('Z'-'A')+1)+'A'; /* Ueberlauf abfangen */
    }

    printf("\nDas neue Wort heisst: %s\n", wort);
}
```

```
    wort[laenge]=(wort[laenge]%('Z'-'A')+1)+'A'; /* Ueberlauf abfangen */
}

printf("\nDas dekodierte Wort heisst: %s\n", wort);
}

/*****
 * strings mit strings addieren (kodieren)
 *
 *      *
 *  Uebung 3, Aufgabe 5c
 *
 *****/

#include<stdio.h>
#include<string.h>

#define MAXSTRING 31

void main(void)
{
    unsigned char wort[MAXSTRING];        /* Deklaration */
    unsigned char code[MAXSTRING];
    unsigned short int wort_zeiger=0, code_zeiger=0;

    printf("\nGib das zu kodierende Wort ein: "); /* Eingabe */
    scanf("%30s", wort);
    printf("\nGib den Code ein: ");
    scanf("%30s", code);

    while (wort[wort_zeiger]!='\0')        /* solange bis wortende erreicht */
    {
        wort[wort_zeiger] += code[code_zeiger]; /* wort und code addieren */
        if (wort[wort_zeiger] > 'Z')          /* Ueberlauf abfangen */
            wort[wort_zeiger] = (wort[wort_zeiger]%('Z'-'A')+1)+'A';
        code_zeiger++;                       /* naechster Buchstabe */
        wort_zeiger++;
        if ('\0' == code[code_zeiger]) code_zeiger=0; /* wenn code zu ende */
    }                                         /* fange wieder vorne an */

    printf("\nDas neue Wort heisst: %s\n", wort);
}
```