

7378 Bytes
13:40

muster.c
1. Oct 1998

```

/*****
 *  Uebung 4, Aufgabe 1
 *  Ampelsimulation
 *****/

#include <stdio.h>

/* Die Ampelfarben */
enum Ampel { rot,rotgelb,gruen,gelb };

int main()
{
    enum Ampel farbe;

    /* Alle Ampelphasen */
    for(farbe=rot;farbe<=gelb;farbe++)
    {
        /* abhaengig von der Farbe eine Nachricht ausgeben */
        switch(farbe)
        {
            case rot:
                printf("Ampel rot: Sofort stehenbleiben!!.\n");
                break;
            case rotgelb:
                printf("Ampel rotgelb: Auf die Plaetze, fertig, ... \n");
                break;
            case gruen:
                printf("Ampel gruen: Und LOS!!!\n");
                break;
            case gelb:
                printf("Ampel gelb: Ohh leck, awei awwer dapper... \n");
                break;
            default:
                printf("ERROR!!\n");
                break;
        }

        sleep(1); /* eine Sekunde warten */
    }

    return 0;
}
```

muster.c
1. Oct 1998

7378 Bytes
13:40

```

/*****
 *  Uebung 4, Aufgabe 2
 *  string <-> int konvertieren
 *****/

#include <stdio.h>

#define MAXSTRINGLAENGE 5

/* Das Menu
   Ausgabe der verschiedenen Wahlmoeglichkeiten und
   Eingabe des Menupunktes.
 *****/
int menu()
{
    int wahl;

    printf("\n\nWillkommen zum gnadenlos guten int <-> string Konvertierer.\n");
    printf("Bitte waehlen sie:\n");
    printf(" (1) int -> string konvertieren\n");
    printf(" (2) string -> int konvertieren\n");
    printf(" (3) Programm beenden\n");
    printf("\n ihre Wahl: ");
    scanf("%d", &wahl);

    return wahl;
}

/* intconvert
   Macht aus einem Integer einen String
   und gibt ihn aus
 *****/
void intconvert(int zahl)
{
    char wort[MAXSTRINGLAENGE];

    /* Die einzelnen Buchstaben durch shiften an die richtige Stelle bringen */
    wort[0] = (char)(zahl>>24);
    wort[1] = (char)(zahl>>16);
    wort[2] = (char)(zahl>>8);
    wort[3] = (char)(zahl>>0);
    wort[4] = '\0'; /* Das Stringende nicht vergessen */

    printf("Die Zahl %d (hex: %08x) wurde zu %s konvertiert.\n", zahl, zahl, wort);
}
```

```
/******  
  stringconvert  
  Macht aus einem String einen Integer  
  und gibt ihn aus  
*****  
void stringconvert(char wort[])          stringconvert  
{  
    int zahl;  
    int wort0=0, wort1=0, wort2=0, wort3=0;          110  
  
    /* Zuerst aus dem Wort einzelne Integer machen */  
    wort0 = (int)wort[0];  
    wort1 = (int)wort[1];  
    wort2 = (int)wort[2];  
    wort3 = (int)wort[3];  
    /* Es hat sich als sehr sinnvoll erwiesen hier eine Ausgabe zum debuggen einzufuegen.  
       Damit kann man ueberpruefen, ob die Integer richtig berechnet wurden:  
  
       printf(“%08x %08x %08x %08x\n”, wort0, wort1, wort2, wort3);          120  
  
       Das Format %08x besagt: gib den integer in heradezimaler Weise (x) aus und zwar  
       mit 8 Zeichen und fuehle links fehlende Zeichen mit einer 0 auf.  
    */  
  
    /* Jetzt die Werte an die richtige Stelle innerhalb des Integer schieben */  
    wort0 <=<= 24;  
    wort1 <=<= 16;  
    wort2 <=<= 8;  
    wort3 <=<= 0;          130  
    /* debug Ausgabe:  
  
       printf(“%08x %08x %08x %08x\n”, wort0, wort1, wort2, wort3);  
  
    */  
  
    /* Zum Schluss alle Integer zu einer Zahl zusammenfassen. */  
    zahl = wort0 | wort1 | wort2 | wort3;  
  
    printf("Der String %s wurde in die Zahl %d (hex: %08x) konvertiert.\n", wort, zahl, zahl);  
}          140  
  
/******  
  Das Hauptprogramm  
*****  
int main()          main  
{  
    char wort[MAXSTRINGLAENGE]; /* das eingelesene Wort */          150
```

```
int zahl;          /* die eingelesene Zahl */  
int wahl=0;          /* Menuepunkt */  
  
while (wahl != 3)  
{  
    /* Zuerst einen Menuepunkt waehlen. */  
    wahl = menu();  
    printf("\n");  
  
    /* Jetzt den entsprechenden Menuepunkt ausfuehren. */          160  
    switch (wahl)  
    {  
        case 1:  
            printf("-----  
            printf("Schoen. Dann geben sie mal bitte einen int ein: ");  
            scanf("%d", &zahl);  
            printf("Ich kovertiere jetzt die Zahl %d in einen string. Einen Moment  
            intconvert(zahl);  
            printf("-----  
            break;          170  
        case 2:  
            printf("-----  
            printf("Gut. Jetzt geben sie bitte einen String ein (4 Zeichen): ");  
            scanf("%4s", wort);  
            printf("Ich kovertiere jetzt den String %s in eine Zahl. Einen Moment.  
            stringconvert(wort);  
            printf("-----  
            break;  
        case 3:  
            printf("\n\n      **** Auf Wiedersehen. ****\n\n\n");          180  
            break;  
        default:  
            printf("\n\n      **** Hoppla. Diesen Menuepunkt bieten wir zur Zeit nicht  
            break;  
    }  
}  
  
return 0;  
}  
  
/******  
  Uebung 4, Aufgabe 3          *  
  Anagramme          *  
*****  
*****  
/******  
  Idee des Programms: Um alle Anagramme eines Wortes der Laenge n zu erhalten  
  haellt man den ersten Buchstaben fest und erzeugt dann alle Anagramme der  
  Laenge n-1; danach vertauscht man bei dem Wort den ersten mit dem zweiten  
  Buchstaben und wiederholt das Ganze; dann vertauscht man den ersten und
```

```
 * den dritten Buchstaben und wiederholt das Ganze ... schliesslich vertauscht *
 * man den ersten mit dem letzten Buchstaben, wiederholt das Ganze und ist fertig. *
 *****/
```

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

```
void ana(char[], int); 210
```

```
int main() main
{
    char wort[21];

    printf("Bitte gebe ein Wort ein (maximal 20 Zeichen lang): ");
    scanf("%20s", wort);
    ana(wort, 0); /* Anagramme werden ab der Position 0 generiert, also alle */
220

    return 1;
}
```

```
void ana(char wort[], int pos) ana
{
    char zwischenspeicher[21], hilf;
    int i;

    strcpy(zwischenspeicher, wort); /* Erst mal das urspruengliche Wort retten */ 230
    for(i=pos; i<strlen(wort); ++i)
    {
        /* Vor jedem neuen Durchlauf muss das alte Wort wieder hergestellt werden */
        strcpy(wort, zwischenspeicher);

        /* Dreieckstausch */
        hilf = wort[pos];
        wort[pos] = wort[i];
        wort[i] = hilf;
240

        /* Falls wir noch nicht am Ende (vorletztes Zeichen) angelangt sind, koennen */
        /* wir noch weitere 'Unteranagramme' erzeugen ... */
        if (pos+1 != strlen(wort)-1)
            ana(wort, pos+1); /* ... und tun dies ... */
        else /* ... ansonsten haben wir das Ende eines Anagramms erreicht ... */
            printf("%s\n", wort); /* ... und geben dieses aus. */
    }
}
```