

```

/*****
/* Titel   : Musterloesungen zum C-Kurs SS 1995 (Ackermann) */
/* Author  : Kilian Heckrodt                               */
/* Inhalt  : Aufgabe 3a                                     */
/* Datum   :                                                */
*****/

/* Sonderzeichen innerhalb des Formatstring muss ein \ vorangestellt sein,
   falls sie als Zeichen ausgegeben werden sollen */

#include <stdio.h>

main() {
    printf("\n So geht's \n", sagter er und freute sich, die Klippen \n");
    printf("der Sonderzeichen umschifft zu haben! \n");
};

/*****
/* Titel   : Musterloesungen zum C-Kurs SS 1995 (Ackermann) */
/* Author  : Kilian Heckrodt                               */
/* Inhalt  : Aufgabe 3b                                     */
/* Datum   :                                                */
*****/

/* Sonderzeichen innerhalb des Formatstring muss ein \ vorangestellt sein,
   falls sie als Zeichen ausgegeben werden sollen */

#include <stdio.h>

main() {
    printf("C:\\prog\\TURBOC.2\\SOURCES\\hugo.c \n");
};

/*****
/* Titel   : Musterloesungen zum C-Kurs SS 1995 (Ackermann) */
/* Author  : Kilian Heckrodt                               */
/* Inhalt  : Aufgabe 4a                                     */
/* Datum   :                                                */
*****/

/* Beispiel 3 */
/* Umwandlung von Fahrenheit in Celsius
   fuer f = 0, 20, ..., 300 */

main()
{
    int lower, upper, step;
    float fahr, celsius;

    lower = 0; /* untere Grenze der Temperaturtabelle */
    upper = 300; /* obere Grenze */
    step = 20; /* Schrittweite */

    printf("Umwandlung von Fahrenheit in Celsius : \n\n");

    fahr = lower;
    while (fahr <= upper) {
        celsius = (5.0/9.0) * (fahr-32.0);
    }
}

```

```

        printf ("%4.0f %6.1f\n", fahr, celsius);
        fahr = fahr + step;
    }
}
/*****
/* Titel   : Musterloesungen zum C-Kurs SS 1995 (Ackermann) */
/* Author  : Kilian Heckrodt                               */
/* Inhalt  : Aufgabe 4b                                     */
/* Datum   :                                                */
*****/

/* Beispiel 3 */
/* Umwandlung von Fahrenheit in Celsius
   fuer f = 0, 20, ..., 300 */

main()
{
    int lower, upper, step;
    float fahr, celsius;

    lower = 0; /* untere Grenze der Temperaturtabelle */
    upper = 300; /* obere Grenze */
    step = 20; /* Schrittweite */

    printf("Umwandlung von Fahrenheit in Celsius : \n\n");
    printf(" Fahrenheit | Celsius  \n");
    printf("----- \n");

    fahr = lower;
    while (fahr <= upper) {
        celsius = (5.0/9.0) * (fahr-32.0);
        printf ("%4.0f      | %6.1f\n", fahr, celsius);
        fahr = fahr + step;
    }
}
/*****
/* Titel   : Musterloesungen zum C-Kurs SS 1995 (Ackermann) */
/* Author  : Kilian Heckrodt                               */
/* Inhalt  : Aufgabe 4c                                     */
/* Datum   :                                                */
*****/

/* Beispiel 3 */
/* Umwandlung von Fahrenheit in Celsius
   fuer f = 0, 20, ..., 300 */

main()
{
    int lower, upper, step;
    float fahr, celsius;

    lower = 0; /* untere Grenze der Temperaturtabelle */
    upper = 300; /* obere Grenze */
    step = 20; /* Schrittweite */

    printf("Umwandlung von Fahrenheit in Celsius : \n\n");
    printf(" Fahrenheit | Celsius  \n");
    printf("----- \n");

    fahr = upper;
    while (fahr >= lower) {
        celsius = (5.0/9.0) * (fahr-32.0);
        printf ("%4.0f      | %6.1f\n", fahr, celsius);

```

```
    fahr = fahr - step;  
}  
}
```

```

/*****
/* Titel   : Musterloesungen zum C-Kurs SS 1995 (Ackermann) */
/* Author  : Kilian Heckrodt                               */
/* Inhalt  : Aufgabe 5                                     */
/* Datum   :                                               */
*****/

```

```

/* Achtung : Die mit (!) versehen Aenderungen sind nicht unbedingt notwendig um
Fehlermeldungen des Compilers zu beseitigen, sondern um ein
"korrektes" Abarbeiten der Befehle zu garantieren */

```

```

/* Beispielprogramm 2 */

```

```

#include <stdio.h> /* falsche include-Datei existiert nur fuer MSDOS */

```

```

main()

```

```

{

```

```

    /* Deklarationsteil */

```

```

    int Num1;
    int num2;
    double realnum;
    char ch;

```

```

    /* Anweisungsteil */

```

```

    Num1 = 3;
    num2 = 66;      /* (!) 66 statt 66., da num2 vom Typ int */
    realnum = 5.5;
    ch = 'A';

```

```

    printf("num1 = %d\n",Num1); /* "Num1" Unterscheidung Gross- und Kleins. */
    printf("num2 = %d\n",num2); /* ";" fehlendes Semikolon und (!) %d statt %f*/

```

```

    printf("realnum = %5.2f\n",realnum);
    printf("ch = %c\n",ch); /* "" fehlendes " am Ende des Formatstrings */

```

```

}

```

```

/*****
/* Titel   : Musterloesungen zum C-Kurs SS 1995 (Ackermann) */
/* Author  : Kilian Heckrodt                               */
/* Inhalt  : Aufgabe 6                                     */
/* Datum   :                                               */
*****/

```

```

#include <stdio.h>

```

```

main() {

```

```

    /* Deklarationsteil */

```

```

    float radius,flaeche,volumen,umfang;
    const float pi = 3.1415927;

```

```

    /* Anweisungsteil */

```

```

    printf("      Radius\t|  Flaeche  \t| Volumen \t|  Umfang\n");
    printf("-----\n");
    /**/

```

```

    /* Achtung wegen der Rundungsfehlern bei Gleitkommazahlen (float, double)
       sollte man Abfragen wie a == 10.0 vermeiden

```

```

        (stattdessen abs(10.0-a) <= 0.001) bzw statt a<= 10.0 besser a<= 10.001*/
    /**/
    for(radius=0.0; radius <= 10.01; radius=radius+0.5) {
        flaeche = pi*radius*radius;
        volumen = 4.0/3.0*pi*radius*radius*radius;
        umfang = 2.0*pi*radius;
        printf("%9.2f \t| %9.2f \t| %9.2f \t| %9.2f \n", radius, flaeche,
                                                       volumen, umfang);
    }; /* for */
}
/*****
/* Titel   : Musterloesungen zum C-Kurs SS 1995 (Ackermann) */
/* Author  : Kilian Heckrodt                               */
/* Inhalt  : Aufgabe 7                                     */
/* Datum   :                                               */
*****/

/* Man beachte : das die Eingabe erst unveraendert auf Bildschirm erscheint
   und erst nach dem ersten Druecken von Return dem Programm uebergeben und
   bearbeitet wird ! */

#include <stdio.h>

void main() {
    char c;
    int ueberlesen;

    c = 's'; /* nur damit c != EOF */
    ueberlesen = 0;
    while (c!= EOF) {
        c=getchar();
        putchar(c);
        while( c == ' ') {
            c=getchar();
            ueberlesen = 1;
        } /* while */
        if (ueberlesen == 1) {
            putchar(c);
            ueberlesen = 0;
        } /* if */
    } /* while */
}

```

```

/*****
/* Titel   : Musterloesungen zum C-Kurs SS 1995 (Ackermann) */
/* Author  : Kilian Heckrodt                               */
/* Inhalt  : Aufgabe 8a                                     */
/* Datum   :                                               */
*****/

/* Man beachte das & vor der Variable in scanf, dies muss dort stehen ! */

#include <stdio.h>

main() {
    int zahl;

    printf("Bitte geben Sie eine Ganzzahl ein : "); scanf("%d",&zahl);
    printf("Sie haben %d eingegeben",zahl);

    return 0; /* Rueckgabewert fuer main (nicht unbedingt noetig) */
};

/*****
/* Titel   : Musterloesungen zum C-Kurs SS 1995 (Ackermann) */
/* Author  : Kilian Heckrodt                               */
/* Inhalt  : Aufgabe 8a                                     */
/* Datum   :                                               */
*****/

/* Man beachte das & vor der Variable in scanf, dies muss dort stehen ! */

#include <stdio.h>

main() {
    int zahl;

    printf("Bitte geben Sie eine Ganzzahl ein : "); scanf("%d",&zahl);
    printf("Sie haben %d eingegeben",zahl);

    return 0; /* Rueckgabewert fuer main (nicht unbedingt noetig) */
};

/*****
/* Titel   : Musterloesungen zum C-Kurs SS 1995 (Ackermann) */
/* Author  : Kilian Heckrodt                               */
/* Inhalt  : Aufgabe 8c                                     */
/* Datum   :                                               */
*****/

/* Man beachte das & vor der Variable in scanf, dies muss dort stehen ! */

#include <stdio.h>

main() {
    int zahl;
    char zeichen;

    printf("Bitte geben Sie eine Ganzzahl und ein Zeichen ein"
           " (Leerzeichen dazwischen) : ");
    scanf("%d %c",&zahl,&zeichen);
    printf("Sie haben %d und %c eingegeben",zahl,zeichen);

    return 0; /* Rueckgabewert fuer main (nicht unbedingt noetig) */
};

```

```

/*****
/* Titel   : Musterloesungen zum C-Kurs SS 1995 (Ackermann) */
/* Author  : Kilian Heckrodt                               */
/* Inhalt  : Aufgabe 8d                                     */
/* Datum   :                                                */
*****/

/* Man beachte das & vor der Variable in scanf, dies muss dort stehen ! */

#include <stdio.h>

main() {
    int zahl;
    char zeichen;

    printf("Bitte geben Sie eine Ganzzahl und ein Zeichen ein"
           " ( - dazwischen) : ");
    scanf("%d-%c",&zahl,&zeichen);
    printf("Sie haben %d-%c eingegeben",zahl,zeichen);

    return 0; /* Rueckgabewert fuer main (nicht unbedingt noetig) */
};

/*****
/* Titel   : Musterloesungen zum C-Kurs SS 1995 (Ackermann) */
/* Author  : Kilian Heckrodt                               */
/* Inhalt  : Aufgabe 8e                                     */
/* Datum   :                                                */
*****/

/* Man beachte das & vor der Variable in scanf, dies muss dort stehen ! */

#include <stdio.h>

main() {
    char string[80];

    printf("Bitte geben Sie eine Zeichenkette ein : ");
    scanf("%s",&string[0]); /* auch string statt &string[0] moeglich */
    printf("Sie haben %s eingegeben",string);

    return 0; /* Rueckgabewert fuer main (nicht unbedingt noetig) */
};

/*****
/* Titel   : Musterloesungen zum C-Kurs SS 1995 (Ackermann) */
/* Author  : Kilian Heckrodt                               */
/* Inhalt  : Aufgabe 9a                                     */
/* Datum   :                                                */
*****/

#include <stdio.h>
#define maxsortenanz 101

main() {
    const float mehrwertsteuer = 0.14;
    float preis[maxsortenanz], gesamtpreis, wert, rabat;
    int artikel[maxsortenanz], sortenanz,i;

    sortenanz = 0;
    while ((sortenanz > maxsortenanz) || (sortenanz == 0)) {

```

```

    printf("Geben Sie die Anzahl der Artikelsorten ein : ");
    scanf("%d",&sortenanz);
} /* while */

printf("\n\n");
for(i=1;i<=sortenanz;i++) {
    printf("Anzahl des Artikels Nr. %d : ",i); scanf("%d",&artikel[i]);
    printf("Preis der Artikels Nr. %d : ",i); scanf("%f",&preis[i]);
    printf("\n");
}; /* for */

printf("\nRabat : "); scanf("%f",&rabat);
printf("\n\n");

for(i=1;i<=sortenanz;i++) {
    wert = ((float) artikel[i])*preis[i];
    gesamtpreis=gesamtpreis + wert;
    printf("Gesamtpreis fuer Artikel Nr. %d : %6.2f\n",i,wert);
}; /* for */

gesamtpreis=gesamtpreis*(1-rabat)*(1+mehrwertsteuer);
printf("Gesamtpreis inklusive Mehrwertsteuer und Rabat : %6.2f \n",
    gesamtpreis);

return 0; /* Rueckgabewert fuer main (nicht unbedingt noetig) */
};

/*****
/* Titel   : Musterloesungen zum C-Kurs SS 1995 (Ackermann) */
/* Author  : Kilian Heckrodt                               */
/* Inhalt  : Aufgabe 9b                                     */
/* Datum   :                                                */
*****/

#include <stdio.h>
#define maxsortenanz 101
#define stringlaenge 80

main() {
    const float mehrwertsteuer = 0.14;
    float preis[maxsortenanz], gesamtpreis, wert, rabat;
    int artikel[maxsortenanz], sortenanz,i;
    char artikelname[maxsortenanz][stringlaenge];

    sortenanz = 0;
    while ((sortenanz > maxsortenanz) || (sortenanz == 0)) {
        printf("Geben Sie die Anzahl der Artikelsorten ein : ");
        scanf("%d",&sortenanz);
    } /* while */

    printf("\n\n");
    for(i=1;i<=sortenanz;i++) {
        printf("Anzahl des Artikels Nr. %d \t: ",i); scanf("%d",&artikel[i]);
        printf("Bezeichnung des Artikels %d \t: ",i); scanf("%s",&artikelname[i]);
        printf("Preis der Artikels Nr. %d \t: ",i); scanf("%f",&preis[i]);
        printf("\n");
    }; /* for */

    printf("\nRabat : "); scanf("%f",&rabat);
    printf("\n\n");

```

```
gesamtpreis = 0;
for(i=1;i<=sortenanz;i++) {
    wert = ((float) artikel[i])*preis[i];
    gesamtpreis=gesamtpreis + wert;
    printf("Gesamtpreis fuer %s \t : %6.2f \n",&artikelname[i],wert);
}; /* for */

gesamtpreis=gesamtpreis*(1-rabat)*(1+mehrwertsteuer);
printf("Gesamtpreis inklusive Mehrwertsteuer und Rabat : %6.2f \n",
    gesamtpreis);

return 0; /* Rueckgabewert fuer main (nicht unbedingt noetig) */
};
```

```

/*****
/* Titel   : Musterloesungen zum C-Kurs SS 1995 (Ackermann) */
/* Author  : Kilian Heckrodt                               */
/* Inhalt  : Aufgabe 10a                                   */
/* Datum   :                                               */
*****/

```

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

```

main() {
    int i,summe,grenze;

    printf("Geben sie bitte die Obergrenze der Summation ein : ");
    scanf("%d",&grenze);
    summe = 0;
    for(i=1;i<=grenze;i++)
        summe += i;    /* summe = summe + i; */
    printf("Summe = %d \n",summe);

    return 0;
};

```

```

/*****
/* Titel   : Musterloesungen zum C-Kurs SS 1995 (Ackermann) */
/* Author  : Kilian Heckrodt                               */
/* Inhalt  : Aufgabe 10b                                   */
/* Datum   :                                               */
*****/

```

```

/* zu beachten : produkt = 1 und nicht produkt = 0 !
   ausserdem wird bei Multiplikation (Fakultaet) schnell der Wertebereich
   der int Variablen ueberschritten,ist dieser einmal ueberschritten
   werden falsche Werte berechnet !
   Das Ueberschreiten des Wertebereichs laesst sich bei int und long meist
   daran erkenne, dass die Zahl ploetzlich negative Werte annimmt, obwohl sie
   positiv sein muesste. Hier ist es sinnvoll statt int eventuell float zu
   verwenden, allerdings erhaelt man dann (wie auf dem Taschenrechner nur
   ein auf 8 Stellen genaues Ergebnis und nicht die exakte Zahl */

```

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

```

main() {
    int i,produkt,grenze;

    printf("Geben sie bitte die Obergrenze der Multiplikation ein : ");
    scanf("%d",&grenze);
    produkt = 1; /* bei Multiplikation muss hier 1 statt 0 stehen ! */
    for(i=1;i<=grenze;i++)
        produkt *= i;    /* produkt = produkt * i; */
    printf("Produkt = %d \n",produkt);

    return 0;
};

```

```

/*****
/* Titel   : Musterloesungen zum C-Kurs SS 1995 (Ackermann) */
/* Author  : Kilian Heckrodt                               */
/* Inhalt  : Aufgabe 11                                    */
/* Datum   :                                                */
*****/

/* Man beachte dass die Speicherbedarf der Datentypen je nach Rechner/
   Betriebssystem/Compiler unterschiedlich sein kann.
   Auf AIX z.B. belegen int und long den gleichen Platz auf MSDOS nicht
*/

```

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

```

main () {
    int a,b,c;
    long d,e,f;
    float g,h,i;
    double j,k,l;
    char m,n,o;
    unsigned char p;

```

```

a = 3.75;
b = 2/3;
d = 1000 * 1000;
c = d;
g = 1/3;
h = 1.0/3.0;
j = 1.0/3.0;
m = 127;
n = m + 1;
o = 255;
p = -1;

```

```

printf("\n");
printf("a: %d\t\tBytes : %d\nb: %d\t\tBytes : %d\nc: %d\t\tBytes : %d\n",
      a,sizeof(a),b,sizeof(b),c,sizeof(c));
printf("d: %d\t\tBytes : %d\ne: %d\t\tBytes : %d\nf: %d\t\tBytes : %d\n",
      d,sizeof(d),e,sizeof(e),f,sizeof(f));
printf("g: %f\t\tBytes : %d\nh: %f\t\tBytes : %d\ni: %f\t\tBytes : %d\n",
      g,sizeof(g),h,sizeof(h),i,sizeof(i));
printf("j: %lf\t\tBytes : %d\nk: %lf\t\tBytes : %d\nl: %lf\t\tBytes : %d\n",
      j,sizeof(j),k,sizeof(k),l,sizeof(l));
printf("m: %c\t\tBytes : %d\nn: %c\t\tBytes : %d\no: %c\t\tBytes : %d\n",
      m,sizeof(m),n,sizeof(n),o,sizeof(o));
printf("p: %c\t\tBytes : %d\n",p,sizeof(p));

```

```
}
```

```

/*****
/* Titel   : Musterloesungen zum C-Kurs SS 1995 (Ackermann) */
/* Author  : Kilian Heckrodt                               */
/* Inhalt  : Aufgabe 12                                    */
/* Datum   :                                                */
*****/

```

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

```
main() {
    char c;

    c = 'c'; /* != EOF */
    while (c!= EOF) {
        c=getchar();
        switch(c) {
            case '\t' : putchar('-'); putchar('>'); break;
            case '\b' : putchar('<'); putchar('-'); break;
            default   : putchar(c);
        }; /* switch */
    }; /* while */
};
```

```

/*****
/* Titel   : Musterloesungen zum C-Kurs SS 1995 (Ackermann) */
/* Author  : Kilian Heckrodt                               */
/* Inhalt  : Aufgabe 13                                     */
/* Datum   :                                                */
*****/

#include <stdio.h>

main() {

    float anfangsschuld,zinsrate,tilgungsrate,annuitaet,restschuld;
    float zinsen,tilgung,gesamtzahlungen,ratio;
    int zaehler,vierteljahr,jahr,start_vierteljahr;

    printf("\nKreditberechnung (alle Angaben in DM) : \n\n");
    printf("Kredit      : "); scanf("%f",&anfangsschuld);
    printf("Zinrsrate   : "); scanf("%f",&zinsrate);
    printf("Tilgungsrate : "); scanf("%f",&tilgungsrate);
    printf("Start       : "); scanf("%d/%d",&start_vierteljahr,&jahr);

    annuitaet =  anfangsschuld*zinsrate/4.0
                + anfangsschuld*tilgungsrate/4.0;
    restschuld = anfangsschuld;
    zaehler = 0;
    vierteljahr = start_vierteljahr - 1;

    printf("Annuitaet      : %8.2f\n\n",annuitaet);
    printf("Zeitraum \t | Zinsen \t | Tilgung\t| Schuldenstand \n");

    while(restschuld >= annuitaet) {
        zaehler++;
        vierteljahr++;
        zinsen    = restschuld * zinsrate/4.0;
        tilgung    = annuitaet - zinsen;
        restschuld = restschuld - tilgung;
        printf("%1d / %4d \t | %8.2f \t | %8.2f \t| %8.2f \n ",
            vierteljahr,jahr,zinsen,tilgung,restschuld);
        if (vierteljahr == 4) {
            jahr++;
            vierteljahr=0;
        } /* if */
    } /* while */

    printf("%1d / %4d \t | %8.2f \t | %8.2f \t| %8.2f \n\n",
        vierteljahr,jahr,zinsen,restschuld,0.0);

    gesamtzahlungen = zaehler * annuitaet + restschuld;
    ratio = gesamtzahlungen/anfangsschuld *100.0;

    printf("Gesamtzahlungen = %8.2f DM\n",gesamtzahlungen);
    printf("Die Gesamtzahlungen betragen %5.2f Prozent des Kredites\n",ratio);
}

```

```

/*****
/* Titel   : Musterloesungen zum C-Kurs SS 1995 (Ackermann) */
/* Author  : Kilian Heckrodt                               */
/* Inhalt  : Aufgabe 14                                     */
/* Datum   :                                                */
*****/

#include <stdio.h>
#include <math.h>

main() {

    float x1_re,x1_im,x2_re,x2_im,p,q,radient;

    printf("Loesung einer quadratischen Gleichung mit der pq-Forme \n");
    printf("Die Gleichung muss die Form x^2+p*x+q=0 haben ! \n");
    printf("p = "); scanf("%f",&p);
    printf("q = "); scanf("%f",&q);

    radient = (p/2.0)*(p/2.0)-q;
    if (radient > 0) {
        x1_re = -p/2.0+sqrt(radient);
        x2_re = -p/2.0-sqrt(radient);
        printf("x1 = %f\n",x1_re);
        printf("x2 = %f\n",x2_re); }
    else {
        x1_im = sqrt(-radient);
        x2_im = -sqrt(-radient);
        x1_re = x2_re = -p/2.0;
        printf("%f\n", -p/2.0);
        printf("x1 = %f + %f*i\n",x1_re,x1_im);
        printf("x2 = %f - %f*i\n",x2_re,-x2_im);
    }
}

```

```

/*****
/* Titel   : Musterloesungen zum C-Kurs SS 1995 (Ackermann) */
/* Author  : Kilian Heckrodt                               */
/* Inhalt  : Aufgabe 15                                     */
/* Datum   :                                                */
*****/

#include<stdio.h>

main()
{
    int a,b;
    a = b = 0;
    /*
    int a = b = 0;
    if a = b
        then printf("a und b sind gleich\n")
        else printf("a und b sind nicht gleich\n");
    if (a <> b)
        printf("a ist ungleich 0\n");
    */
    if (a==b)
        printf("a und b sind gleich\n");
}

```

```
else
    printf("a und b sind nicht gleich\n");
if(a!=0)
    printf("a ist ungleich 0\n");
}
```

```

/*****
/* Titel   : Musterloesungen zum C-Kurs SS 1995 (Ackermann) */
/* Author  : Kilian Heckrodt                               */
/* Inhalt  : Aufgabe 16                                     */
/* Datum   :                                                */
*****/

/* Es werden alle gerade Zahlen bis 10 ausgegeben.
   i % 2 ergibt 0 , wenn die Zahl durch 2 teilbar ist, 0 bei if
   bedeutet als logischer Wert jedoch falsch (d.h. if-Bedingung nicht
   erfuehlt), also wird der else-Teil ausgefuehrt.
   liefert i % 2 irgendetwas ungleich null so wird dieses als wahr
   (if-Bedingung erfuehlt betrachtet)
*/

```

```

#include <stdio.h>

```

```

main(){
    int i = 1;

    while (i++ < 10)
        if (i % 2)
            continue;
        else
            printf("%d ",i);
}

```

```

/*****
/* Titel   : Musterloesungen zum C-Kurs SS 1995 (Ackermann) */
/* Author  : Kilian Heckrodt                               */
/* Inhalt  : Aufgabe 17                                     */
/* Datum   :                                                */
*****/

/* Hinweis : Man beachte unbedingt die Formatanweisungen bei printf
              und scanf, stimmen diese nicht, so fuehrt es in der Regel
              zu voellig unsinnigen Ausgaben/ergebnissen.
              fuer double verwende man %lf

*/

```

```

#include <stdio.h>

```

```

int sumpar1(int zahl1, int zahl2) ;
float sumpar2(float zahl1, float zahl2);

```

```

main(){
    int z1,z2;
    float r1,r2;

    printf("\n");
    printf("1. Integerzahl : "); scanf("%d",&z1);
    printf("2. Integerzahl : "); scanf("%d",&z2);
}

```

```

printf("1. Floatzahl   : "); scanf("%f",&r1);
printf("2. Floatzahl   : "); scanf("%f",&r2);

printf("1. Summe = %d\n",sumpar1(z1,z2));
printf("2. Summe = %f\n",sumpar2(r1,r2));

};

int sumpar1(int zahl1, int zahl2) {
    return zahl1 + zahl2;
};

float sumpar2(float zahl1, float zahl2) {
    return zahl1 + zahl2;
};

/*****
/* Titel   : Musterloesungen zum C-Kurs SS 1995 (Ackermann) */
/* Author  : Kilian Heckrodt                               */
/* Inhalt  : Aufgabe 18                                     */
/* Datum   :                                                */
*****/

int j = 6;

main() {
    int i = 3;
    printf("%d ",i); /* i = 3 */
    printf("%d ",j); /* j = 6 */
    {
        int i = 6;
        printf("%d ",i); /* i = 6 das i von oberhalb ist hier ueberdeckt !*/
        printf("%d ",j); /* j = 6 j nicht ueberdeckt !*/
        j = 0;
        i += 2;
    }
    printf("%d ",i); /* i = 3 wieder altes i (wurde nicht veraendert) */
    printf("%d ",j); /* j = 0 wurde veraendert */
}

```

```

/*****
/* Titel   : Musterloesungen zum C-Kurs SS 1995 (Ackermann) */
/* Author  : Kilian Heckrodt                               */
/* Inhalt  : Aufgabe 19                                     */
/* Datum   :                                                */
*****/

/* Das Programm erzeugt die Ausgabe 7 6 5 4 3 2 1
   Die Zahlenerscheinen in dieser Reihenfolge, weil die Ausgabe
   erst nach dem rekursiven Aufruf geschieht, d.h. printf wird
   zum ersten mal erreicht bei x = 7 nachdem rekurs(x+1) ausgefuehrt
   worden ist. Danach springt die Funktion auf die naechste hoehere
   Aufrufebene zurueck (da fuer x = 7 rekurs nun abgearbeitet ist),
   in dieser ist x = 6 nun wird dann printf ausgefuehrt und 6 ausgegeben
   usw.
*/

```

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

```
void rekurs(int i);
```

```
main(){
    rekurs(1);
}
```

```
void rekurs(int x) {
    if (x == 8)
        return;
    else
        rekurs(x+1);
    printf("%d ",x);
}
```

```

/*****
/* Titel   : Musterloesungen zum C-Kurs SS 1995 (Ackermann) */
/* Author  : Kilian Heckrodt                               */
/* Inhalt  : Aufgabe 20  LOesung a) verwendet Aufgabe 19     */
/* Datum   :                                                */
*****/

```

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

```
void reverse(char wort[], int x);
```

```
main(){
    reverse("ein neger mit gazelle zagt im regen nie",0);
}
```

```
void reverse(char wort[], int x) {
    if (x == strlen(wort))
        return;
    else
        reverse(wort,x+1);
    printf("%c",wort[x]);
}
```

[illegible]

```
void belegen(float quadrat[],int anzahl);
void ausgeben(float quadrat[],int anzahl);
```

```
main(){
    float feld[100];

    belegen(feld,100);
    ausgeben(feld,100);
```

```
}
```

```
void belegen(float quadrat[],int anzahl) {

    int i;

    for(i=0;i<=anzahl-1;i++)
        quadrat[i]=i*i;
}
```

```
void ausgeben(float quadrat[],int anzahl) {

    int i;
    printf("\nQuadratzahlen : \n");
    for(i=0;i<=anzahl-1;i++) {
        printf("%6.0f\t",quadrat[i]);
        if ((i+1) % 8 == 0)
            printf("\n");
    }; /* for */
}
```

```

/*****
/* Titel   : Musterloesungen zum C-Kurs SS 1995 (Ackermann) */
/* Author  : Kilian Heckrodt                               */
/* Inhalt  : Aufgabe 23 Loesung a) ohne Rekursion           */
/* Datum   :                                                */
*****/
```

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

```
void sortiere(int feld[], int anzahl);
int min_pos(int feld[],int startpos,int anzahl);
void vertausche(int *a, int *b);
void ausgabe(int feld[], int anzahl);
```

```
main() {
    int feld[] = {23,12,45,26,76,88,123,2,34,56,0,98,45,64,23,78,
                  8,82,50,20};

    sortiere(feld,20);
    ausgabe(feld,20);
}
```

```

void ausgabe(int feld[], int anzahl) {
    int i;

    printf("\naufsteigend sortiert : \n");
    for(i=0;i<=anzahl-1;i++){
        printf("%d\t",feld[i]);
        if((i-1) % 8 == 0)
            printf("\n");
    }
}

void sortiere(int feld[], int anzahl){
    int i, j;

    for(i=0;i<=anzahl-1;i++) {
        j=min_pos(feld,i,anzahl);
        vertausche(&feld[i],&feld[j]);
    };
}

int min_pos(int feld[],int startpos,int anzahl) {
    int ergebnis, i;

    ergebnis = startpos;
    for(i=startpos;i<=anzahl-1;i++)
        if (feld[i] < feld[ergebnis])
            ergebnis = i;
    return ergebnis;
}

void vertausche(int *a, int *b) {
    int hilf;

    hilf = *a;
    *a = *b;
    *b = hilf;
}

/*****
/* Titel   : Musterloesungen zum C-Kurs SS 1995 (Ackermann) */
/* Author  : Kilian Heckrodt                               */
/* Inhalt   : Aufgabe 23 Loesung b) mit Rekursion           */
/* Datum    :                                               */
*****/

#include <stdio.h>

void hilf_sortiere(int feld[], int anzahl, int i);
void sortiere(int feld[], int anzahl);
int min_pos(int feld[],int startpos,int anzahl);

```

```

void vertausche(int *a, int *b);
void ausgabe(int feld[], int anzahl);

main() {
    int feld[] = {23,12,45,26,76,88,123,2,34,56,0,98,45,64,23,78,
                  8,82,50,20};

    sortiere(feld,20);
    ausgabe(feld,20);
}

void ausgabe(int feld[], int anzahl) {

    int i;

    printf("\naufsteigend sortiert : \n");
    for(i=0;i<=anzahl-1;i++){
        printf("%d\t",feld[i]);
        if((i-1) % 8 == 0)
            printf("\n");
    }
}

void sortiere(int feld[], int anzahl){

    hilf_sortiere(feld,anzahl,0);
}

void hilf_sortiere(int feld[], int anzahl, int i) {

    int j;

    if(i == anzahl-1)
        return;
    else {
        j=min_pos(feld,i,anzahl);
        vertausche(&feld[i],&feld[j]);
        hilf_sortiere(feld,anzahl,i+1);
    } /* if - else */
};

int min_pos(int feld[],int startpos,int anzahl) {

    int ergebnis, i;

    ergebnis = startpos;
    for(i=startpos;i<=anzahl-1;i++)
        if (feld[i] < feld[ergebnis])
            ergebnis = i;
    return ergebnis;
}

void vertausche(int *a, int *b) {

    int hilf;

```

```
    hilf = *a;  
    *a = *b;  
    *b = hilf;  
}
```