

```
/* ***** */
/* Musterl"osung "Ubung 2, Aufgabe 1 */
/* ***** */

#include<stdio.h>

/* ----- */
/* Funktionsdeklaration */
/* ----- */
/* die Funktion arith1: sie liefert das arithmetische Mittel zur"uck */
float arith1(int a, int b) 10
{
    return ( ((float)(a+b)) / 2);
}
/* ANMERKUNG:
   Die Variablen a und b dieser Funktion sind lokale Variablen und somit
   nicht dieselben, wie die des Hauptprogramms, d.h. wenn man in arith1
   den Wert von a ver"andert, so hat dies keine Auswirkungen auf den Wert
   des a's aus dem Hauptprogramm. 20
   Testen sie dies, indem sie in arith1 das a veraendern (zB: a=42;) und
   im Hauptprogramm das a vor und nach dem Aufruf von arith1 ausgeben.
   Ausserdem k"onnte man noch das return anderes schreiben
   mit implizitem Cast:
       return ( (a+b)/2.0) ;
*/

/* die Funktion airth2: sie gibt das arithmetische Mittel aus */
void arith2(int a, int b) 30
{
    printf("arith2: Das arithmetische Mittel von %d und %d ist %f.\n",
        a, b, (float)(a+b)/2);
}

/* ----- */
/* Das Hauptprogramm */
/* ----- */
int main (void) 40
{
    /* Variablendeklaration */
    int a;
    int b;
    float mittel;

    /* Einlesen der Variablen */
    printf("Bitte gebe eine ganze Zahl ein: ");
    scanf("%d", &a);
    printf("Bitte gebe eine ganze Zahl ein: ");
    scanf("%d", &b); 50
}
```

```
/* Aufruf von arith1() */
mittel = arith1(a,b);
printf("arith1: Das arithmetische Mittel von %d und %d ist %f.\n",
    a, b, mittel);

/* Aufruf von arith2() */
arith2(a,b);

return 0; /** R"uckgabewert ans Betriebssystem */ 60
}

/* ***** */
/* Musterl"osung "Ubung 2, Aufgabe 2 */
/* ***** */

#include<stdio.h>

/* ----- */
/* gloable Varibalen */
/* ----- */
float a;
float b;

/* ----- */
/* Funktionsdeklaration */
/* ----- */
/* die Funktion min1: sie bekommt zwei float Werte als Argumente */
float min1(float a, float b) 70
{
    /* Versuch: "Andern sie hier die Werte von a und b und pr"ufen sie,
       was mit den globalen Variablen a und b passiert.
    */
    if (a < b) return a;
    return b;
}

/* die Funktion min2: sie hat keine Argumente */
float min2() 80
{
    if (a < b) return a;
    return b;
}

/* ----- */
/* Das Hauptprogramm */
/* ----- */ 90
/* 100
```

```
/* ----- */
int main (void)                                main
{
    /* Variablendeklaration */
    float minimum;

    /* Einlesen der Variablen */
    printf("Bitte gebe eine Fließkommazahl ein: ");
    scanf("%f", &a);
    printf("Bitte gebe noch eine Fließkommazahl ein: ");    110
    scanf("%f", &b);

    /* Aufruf von min1() */
    minimum = min1(a,b);
    printf("min1: Das Minimum von %f und %f ist %f.\n",
        a, b, minimum);

    /* Aufruf von min2() */
    minimum = min2();
    printf("min2: Das Minimum von %f und %f ist %f.\n",    120
        a, b, minimum);

    return 0; /* R "uckgabewert ans Betriebssystem */
}

/* ***** */
/* Musterlösung "Ubung 2, Aufgabe 2" */
/* ***** */
#include<stdio.h>                                130

/* ----- */
/* Funktionsdeklaration                        */
/* ----- */
/* iterative Berechnung der Fakultät */
int iter_faq(int n)                                iter_faq
{
    int faq=1;                                140

    while (n>=1)
    {
        faq = faq * n;
        n--;
    }
    /* oder superkurz und superunübersichtlich: while (n) faq *= n--; */

    return faq;
}                                150
```

```
/* rekursive Berechnung der Fakultät */
int rek_faq(int n)                                rek_faq
{
    if (n==1) return 1;                        /* Rekursionsende bei 1 */
    if (n > 1) return n*rek_faq(n-1); /* Rekursionsschritt wenn n groesser als 1 */
    return 1;                                /* Falls n<=0 ist gib eine 1 zuruck */
}                                160

/* ----- */
/* Das Hauptprogramm */
/* ----- */
int main (void)                                main
{
    /* Variablendeklaration */
    int n;
    int fakultaet;                                170

    /* Einlesen der Variablen */
    printf("Bitte gebe eine natuerliche Zahl ein: ");
    scanf("%d", &n);

    /* Aufruf von iter_faq() */
    fakultaet = iter_faq(n);
    printf("iter_faq: Die Fakultät von %d ist %d.\n", n, fakultaet);

    /* Aufruf von rek_faq() */
    fakultaet = rek_faq(n);                                180
    printf("rek_faq: Die Fakultät von %d ist %d.\n", n, fakultaet);

    return 0; /* R "uckgabewert ans Betriebssystem */
}
```