```c
/* ----------------------------------- */
/* Musterloesung Uebung 1, Aufgabe 1                          */
/* ----------------------------------- */
#include<stdio.h>

int main (void)                                          main
{
  /* Variablendeklaration */
  int a;
  float b=7.1;

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

  /* Ausgabe */
  printf("\nDie Eingabe war wie folgt: a=%d, b=%f\n", a, b);
  return 0;  /*** Rueckgabewert ans Betriebssystem ***/
}


/* ------------------------- */
/* Musterloesung Uebung 1, Aufgabe 2                 */
/* ------------------------- */

#include<stdio.h>

int main (void)                                          main
{
  int n, i;

  printf("Bitte gebe eine natuerliche Zahl ein: ");
  scanf("%d", &n);

  /* Die Schleife wird n-mal durchlaufen */
  i=1;
  while (i<=n)
  {
    /*** '\t' fuegt einen Tabulator ein ***/
    printf("%d :\t %d\n", i, i*i);
    i=i+1;
  }

  return 0;
}


/* ----------------------- */
```



```c
/* Musterloesung Uebung 1, Aufgabe 3                 */
/* ------------------------ */

#include<stdio.h>

int main (void)                                          main
{
  int x, y;

  printf("Bitte gebe eine natuerliche Zahl ein: ");
  scanf("%d", &x);
  printf("Bitte gebe noch eine natuerliche Zahl ein: ");
  scanf("%d", &y);

  if (x<y)
    printf("%d < %d\n", x, y);
  else if (x>y)
    printf("%d > %d\n", x, y);
  else
    printf("%d = %d\n", x, y);

  return 0;
}


/* --------------------------- */
/* Musterloesung Uebung 1, Aufgabe 4                  */
/* --------------------------- */

#include<stdio.h>

int maximum (int,int);   /* Deklaration der Funktion maximum (Funktions-Prototyp) */

/**********************************
*  Hier beginnt das Hauptprogramm  *
**********************************/
int main (void)                                          main
{
  int x ,y;

  printf("Bitte gebe eine natuerliche Zahl ein: ");
  scanf("%d", &x);
  printf("Bitte gebe noch eine natuerliche Zahl ein: ");
  scanf("%d", &y);

  /* Ausgabe mit Aufruf der Funftion maximum */
  printf("Das Maximum von %d und %d ist %d\n", x, y, maximum(x,y));

  return 0;
}
```

```c
/**************************************************************
*  Hier beginnt die Funktion maximum.                         *
*  Sie bekommt 2 int-Werte und gibt das Maximum davon zurueck. *
**************************************************************/
int maximum(int a, int b)                                          maximum
{
  if (a<b)
    return b;
  return a;   /*** Diese Zeile wird nur ausgefuehrt, falls a>=b ***/
}


/* ---------------------------------------- */
/* Musterloesung Uebung 1, Aufgabe 5        */
/* ---------------------------------------- */

#include<stdio.h>

int prim (int);          /* Funktions-Prototyp */

int main (void)                                                    main
{
  int i, n;

  printf("Bis zu welcher Zahl willst Du die Primzahlen wissen? ");
  scanf("%d", &n);

  i=2;     /*** 1 ist sowieso keine Primzahl ***/
  while (i<=n)
  {
    if(prim(i))
      printf("%d ist eine Primzahl.\n", i);
    ++i;
  }

  return 0;
}

/********************************************************************
*  Hier beginnt die Funktion prim.                                  *
*  Sie bekommt einen int-Wert und gibt 1 (WAHR) zurueck, wenn diese *
*  Zahl eine Primzahl ist und sonst gibt sie 0 (FALSCH) zurueck.    *
********************************************************************/
int prim(int x)                                                    prim
{
  int i;

  /*** Es wird versucht, einen Teiler von x zu finden. Wird keiner ***/
  /*** gefunden, so ist die Zahl prim.  Es reicht zu testen, ob    ***/
```



```c
  /*** die Zahlen von 2 bis x/2 Teiler von x sind oder nicht.      ***/
  i=2;
  while(i <= x/2)
  {
    if (x%i == 0)   /*** x modulo i ist gleich 0, also ist i ein Teiler von x ***/
      return 0;     /*** und deshalb x keine Primzahl.                        ***/
    ++i;
  }
  return 1;         /*** Es wurde kein Teiler gefunden und deshalb ist x prim. ***/
}
```