

## Kapitel 8 Beispiel 12

```
1 // Programm 8.3.1 Intervallhalbierungsmethode.cpp
2 // Programm zur Berechnung von Nullstellen
3 // nach dem Intervallhalbierungsverfahren
4 // Autor: Heiderich / Meyer
5 // -----
6 #include "stdafx.h"
7 #include <stdio.h>
8 #include <stdlib.h>
9 #include <conio.h>
10 #include "signum.h"
11 #include "f.h"
12 #include <math.h>
13 using namespace System;
14 void main()
15 {
16     float xu, xo, epsilon, xm;
17     int vz;
18     // Begrüßung
19     printf("\n\n\tProgramm zur Berechnung von Nullstellen\n");
20     printf("\tnach dem Intervallhalbierungs-Verfahren\n");
21     printf("\t-----\n");
22     // Eingabe der Intervallgrenzen und der Genauigkeit
23     printf("\n\tBitte geben Sie die untere Intervallgrenze      xu
24         an: ");
25     fflush(stdin);
26     scanf("%f",&xu);
27     printf("\n\tBitte geben Sie die obere Intervallgrenze      xo
28         an: ");
29     fflush(stdin);
30     scanf("%f",&xo);
31     printf("\n\tBitte geben Sie die gewünschte Genauigkeit epsilon
32         an: ",char(129));
33     fflush(stdin);
34     scanf("%f",&epsilon);
35     printf("\n");
36     do
37     {
38         xm = ( xo + xu ) / 2.;
39         if ( signum( f(xm) ) == signum ( f(xu) ) ) xu = xm;
40         else xo = xm;
41         printf("\n\txu = %10.6f  xo = %10.6f\n\n",xu,xo);
42     } while ( abs(xu-xo) > epsilon || f(xm) != 0.0);
43     // Ausgabe Ergebnisse
44     printf("\n\tdie Funktion besitzt eine Nullstelle zwischen\n");
45     printf("\n\txu = %f    und    xo = %f\n",xu,xo);
46     getch();
47 }
```