

```

1  import ProgTools.IOTools;
2  public class Eval2 {
3      public static void main(String[] args) { // Hauptprogramm
4          int n = IOTools.readInteger("n="); // lies n ein
5          double l = IOTools.readDouble("l="); // lies l ein
6          double r = IOTools.readDouble("r="); // lies r ein
7
8          double produkt = 1.0; // Berechnung der Potenz
9          for (int i=0; i < 2*n; i++) // ...
10             produkt = produkt * l; // abgeschlossen
11          double f_l_n = produkt + n*n - n*l; // Berechnung von f
12          System.out.println("f(l,n)=" + f_l_n); // Ergebnisausgabe
13
14          produkt = 1.0; // Berechnung der Potenz
15          for (int i=0; i < 2*n; i++) // ...
16             produkt = produkt * r; // berechnet
17          double f_r_n = produkt + n*n - n*r; // Berechnung von f
18          System.out.println("f(r,n)=" + f_r_n); // Ergebnisausgabe
19
20          double m = (l + r) / 2.0; // Mittelpunkt
21          produkt = 1.0; // Berechnung der Potenz
22          for (int i=0; i < 2*n; i++) // ...
23             produkt = produkt * m; // abgeschlossen
24          double f_m_n = produkt + n*n - n*m; // Berechnung von f(M,n)
25          System.out.println("f(m,n)=" + f_m_n); // Ergebnisausgabe
26
27          double mw = (f_l_n + f_r_n + f_m_n)/3; // Mittelwert
28          System.out.println("Mittelwert=" + mw); // Ergebnisausgabe
29      }
30  }

```

Wir sehen, dass unser neues Programm wesentlich länger und leider auch unübersichtlich geworden ist. Der Grund hierfür liegt vor allem an der sich ständig wiederholenden **for**-Schleife. Leider benötigen wir diese aber für die Berechnung der Funktion f . Zu schade, dass wir diese nicht wie den Sinus oder Tangens als einen eigenständigen Befehl zur Verfügung stellen können! Oder etwa doch?

In den folgenden Abschnitten lernen wir, so genannte **Methoden** (oder auch **Routinen**) zu definieren. Dies sind Unterprogramme, die vom Hauptprogramm (der **main**-Methode) aufgerufen werden und auch Ergebnisse zurückliefern können. Mit ihrer Hilfe werden wir Programme schreiben, die weit komplexer als obiges Beispiel, aber dennoch übersichtlicher sind!

6.1 Methoden

6.1.1 Was sind Methoden?

Durch Methoden wird ausführbarer Code unter einem Namen zusammengefasst. Dieser Code kann unter Verwendung so genannter Parameter formuliert sein, denen später beim Aufruf der Methode Werte übergeben werden. Wie im Abschnitt über Klassen bereits erwähnt, gehören Methoden in der Regel neben den Varia-