Aufgabe 1. Implementiere eine Funktion die zu einem gegebenen Funktionenpointer $f : \mathbb{R} \to \mathbb{R}$, einem Dateinamen, einer Schrittweite $s \in \mathbb{R}$, einer Startstelle x_1 und einer Endstelle x_2 die Wertetabelle der Funktion zwischen x_1 und x_2 zur Schrittweite s speichert. Dabei sollen x und $f(x)$ durch einen Tabulator getrennt werden und jedes Paar $(x, f(x))$ in einer eigenen Zeile stehen. Etwa wäre die Ausgabe für $f = \cos$ zwischen $x_1 = 0$ und $x_2 = 0$ mit Schrittweite $s = 0.1$ die folgende:

```
0.0	1.0
0.1	0.995004165278
0.2	0.980066577841
0.3	0.955336489126
0.4	0.921060994003
0.5	0.87758256189
0.6	0.82533561491
0.7	0.764842187284
0.8	0.696706709347
0.9	0.621609968271
1.0	0.540302305868
```

Aufgabe 2. In dieser Aufgabe geht es um numerische Integration.

a) Implementiere eine Integrationsfunktion, die das Intervall $[a, b]$ in n gleich große Teile aufteilt, für diese jeweils die Trapezsumme (aus der Vorlesung) berechnet und diese aufsummiert:

```c
double integrate(double a, double b,
  double (*f)(double), unsigned int n);
```

b) Schreibe nun eine Funktion, die nicht die Anzahl der Teilintervalle erhält, sondern eine "Fehlertoleranz" e. Die Funktion die Aufteilung solange verfeinern, bis sich der approximierte Wert für das Integral durch eine Verfeinerung nur noch um weniger als e ändern würde.