Übungen zu Physik auf dem Computer SS 2012

Übungsblatt 8: Integration

13. Juni 2012

Allgemeine Hinweise

- Abgabetermin ist **Montag, 18.6.2012, 13:00**
- Zur Abgabe schickst Du die Lösungsdatei(en) im Anhang einer Email an Deinen Tutor:
  - Florian (`floh@icp.uni-stuttgart.de`; Dienstag, 15:45–17:15)
  - Dominic (`dominic@icp.uni-stuttgart.de`; Dienstag, 15:45–17:15)
  - Olaf (`olenz@icp.uni-stuttgart.de`; Mittwoch, 15:45–17:15)
- Die Übungen werden in Gruppen von jeweils zwei oder drei Leuten bearbeitet. Diese dürfen sich gerne von Blatt zu Blatt unterscheiden. Aus formalen Gründen muss allerdings jeder von Euch eine eigene Lösung abgeben. Schreibt bitte auf die Lösungen, mit wem Ihr zusammengearbeitet habt, um uns das Korrigieren zu erleichtern.
- Die Übungen finden statt im CIP-Pool des Instituts für Computerphysik (ICP) im Pfaffenwaldring 27.

Aufgabe 8.1 (7 Punkte): Kanonenkugel

Aufgabe 8.1.1 (4 Punkte): Auftreffort

Die Trajektorie einer Kanonenkugel $\mathbf{r}(t)$ ist bestimmt durch Anfangsort $\mathbf{r}_0$ und Anfangsgeschwindigkeit $\mathbf{v}_0$. Es wirken die Luftreibung und die Gravitation als Kräfte auf die Kugel.

Die Bewegungsgleichungen der Kugel lauten entsprechend:

$$\mathbf{F}(t) = \ddot{\mathbf{r}}(t) = \frac{\mathbf{F}_R(t, v(t))}{m} - \begin{pmatrix} 0 \\ g \end{pmatrix} \tag{1}$$

$$\mathbf{r}(t) = \int \mathbf{v}(t)dt = \int dt \int \mathbf{F}(t)dt \tag{2}$$

mit der Reibungskraft $\mathbf{F}_R$, der Masse m und der Erdbeschleunigung g. Als Randbedingungen sind $\dot{\mathbf{r}}(t_0) = \mathbf{v_0}$ und $\mathbf{r}(t_0) = \mathbf{0}$ sinnvoll, um die Integrationskonstanten zu bestimmen. Die bereits vorgegebene Funktion `speed(t)` in der Datei `/share/Courses/PC2012/canonspeed.py` gibt die Geschwindigkeit $\dot{\mathbf{r}}$ der Kugel zum Zeitpunkt t nach dem Abschuss (für eine Default Anfangsgeschwindigkeit) als x-y-Tupel zurück. Die Funktion arbeitet eher langsam und darf hier als „black box“ verstanden werden.

- (2 Punkte) Erstelle ein Python-Skript `cannonball.py` und importiere die Datei mit der Geschwindigkeitsfunktion. Implementiere die Methode „Zusammengesetzte Mittelpunktsregel“, um das Geschwindigkeitsintegral der Gleichung (2) zu lösen. Die Anzahl der Stützstellen sollte als Parameter übergeben werden können. Gib das fertige Skript als Lösung ab.

- (2 Punkte) Bestimme in einer Schleife den Auftreffpunkt der Kugel ($y_a = 0$). Wähle dabei als Schrittweite für die Zeit $dt = 0.02$. Plotte mit Pyplot und diskutiere den Auftreffort für (4, 8, 16, 32, 64) Stützstellen. Gib das fertige Skript als Lösung ab.

Aufgabe 8.1.2 (3 Punkte): Romberg-Integration

- (2 Punkte) Erweitere das Skript, so dass es zusätzlich eine Romberg-Integration ausführt. Verwende nur die drei ersten Stützstellen (4, 8, 16) für die Interpolation. Gib das erweiterte Skript als Lösung ab.
- (1 Punkt) Vergleiche das Ergebnis und die Laufzeit der beiden Methoden. Was erscheint schneller? Was erscheint genauer? Schreibe die Antworten in die eMail an deinen Tutor.

Aufgabe 8.2 (3 Punkte): Monte-Carlo Integration

- (2 Punkte) Berechne in einem Python-Skript unter Verwendung der Monte-Carlo-Methode folgendes Integral:
$$\int_0^1 \int_0^1 \int_0^1 e^{x^2+y^2+z^2} dxdydz \tag{3}$$
Gib das fertige Skript als Lösung ab.
- (1 Punkt) Verwende die fertige Funktion `scipy.integrate.tplquad`, um einen Referenzwert für das oben stehende Integral zu berechnen. Berechne jeweils 20 Mal das Integral mit der Monte-Carlo-Methode mit 10^i Punkten ($i = 1 \dots 6$)). Plotte mit Pyplot jeweils die mittlere Abweichung vom Referenzwert aus diesen 20 Berechnungen über die Anzahl der Punkte in einer doppelt logarithmischen Skala. Was erwartest Du, welche Steigung der entstehende Graph hat? Welche Methode verwendet (laut Dokumentation) `tplquad` für die Berechnung? Gib das veränderte Skript als Lösung ab und schreibe die Antworten in die Lösungsmail.