Praktikum zur Numerischen Mathematik im Sommersemester 2021
Prof. Dr. Nils Rosehr, Hochschule München, 14. April 2021

# Praktikum 1: Numerische Integration

## Mathematische Vorüberlegungen

Berechnen Sie für eine Funktion $f : [a, b] \to \mathbb{R}$ das Integral

$$\int_a^b f(x)\,dx$$

exakt per Hand für folgende Angaben:

(f1) $a = -1, \quad b = 2, \quad f(x) = x^3$ (siehe auch Aufgabe 2.3).

(f2) $a = 0, \quad b = 2, \quad f(x) = x^4 - 2x^3 + 3$ (siehe auch Aufgabe 2.5).

(f3) $a = 0, \quad b = 32, \quad f(x) = 3\sqrt[5]{x}$.

(f4) $a = 0, \quad b = 7\pi, \quad f(x) = \sin(x)$.

## Implementieren einer Bibliothek

- Informieren Sie sich über das Prinzip DRY in der Programmierung und behalten Sie Ihre Erkenntnisse im Hinterkopf bei den folgenden Aufgaben dieses ersten Praktikums – und auch für alle weiteren Praktika.
- Schreiben Sie eine Bibliothek `lib1`, d.h. eine Datei `lib1.py`, in der (mindestens) folgende Funktionen definiert sind:
  - `R(f, a, b), T(f, a, b), S(f, a, b)`
  - `Rh(f, a, b, n), Th(f, a, b, n), Sh(f, a, b, n).`

  Dabei sei $f$ eine reellwertige Funktion, die mindestens auf dem Intervall $[a, b]$ definiert ist, und $n$ eine natürliche Zahl. Die Funktionen `R`, `T` und `S` sollen die Rechteck-, Trapez- und Simpson-Regel implementieren, und die Funktionen `Rh`, `Th` und `Sh` sollen die summierten Varianten dieser Näherungen mit $n$ Teilintervallen gleicher Breite sein. Rückgabewerte (mit `return`) dieser Funktionen sollen die Näherungswerte der entsprechenden Integrationsmethoden sein.

## Testen der Bibliothek

- Schreiben Sie ein Programm `main1.py`, in dem Sie Funktionen `f1`, `f2`, `f3` und `f4` implementieren, die obige Funktionen berechnen. Zur Berechnung der Funktion sollten Sie in Hinblick auf den folgenden Punkt die Bibliothek `numpy` der Bibliothek `math` vorziehen.
- Plotten Sie diese, indem Sie etwa

```
import numpy as np
import matplotlib.pyplot as plt
x = np.linspace(-1, 2, 100)    # 100 Stützstellen
plt.plot(x, f1(x))             # Plot von -1 bis 2
plt.show()                     # Plot anzeigen
```

  in `main1.py` einfügen.
- Definieren Sie in `main1.py` eine weitere Funktion `true_result(f)`, die als Argument eine der Funktionen `f1`, `f2`, `f3` oder `f4` annimmt und das exakte Ergebnis ihrer händischen Berechnungen von oben zurück gibt. (Am elegantesten geht dies mit einem Dictionary.)
- Testen Sie nun Ihre Bibliothek von nummerischen Integrationsmethoden, indem Sie in `main1.py` jeweils die vier Integrale für verschiedene Anzahlen von Teilintervallen berechnen und mit den exakten Werten vergleichen.
- Einige Fragen, die Sie nun beantworten können sollten:
  - Wieviele Teilintervalle sind jeweils nötig, um einen absoluten Fehler von höchstens $1/10$ zu erhalten.
  - Was fällt bei `f1`, `f3` und `f4` auf? Und woran liegt das?
- Ermitteln Sie – z.B. durch Probieren – die kleinste Anzahl $n_i$ von Teilintervallen, die für die Funktion `fi` von oben nötig ist, um einen absoluten Fehler von höchstens $1/10$ bei der summierten Simpson-Regel zu erhalten. Definieren Sie nun in `main1.py` eine Liste `Anzahlen`, so dass `Anzahlen[i-1]` die ermittelte Anzahl $n_i$ ist.

## Anwendung

Eine wichtige Wahrscheinlichkeitsverteilung in der Mathematik sowie naturwissenschaftlichen und technischen Anwendungen ist die Gaußsche Normalverteilung. Wir wollen hier die Standard-Normalverteilung mit der Dichtefunktion $\varphi(x) = e^{-x^2/2}/\sqrt{2\pi}$ für $x \in \mathbb{R}$ betrachten. Für eine standard-normalverteilte Messgröße beträgt die Wahrscheinlichkeit, dass ein einzelnes Messergebnis im Intervall $[a, b]$ liegt, $\int_a^b \varphi(x)\,dx$. Leider lässt sich dieses Integral nicht für beliebige Grenzen $a$, $b$ exakt berechnen, so dass numerische Methoden herangezogen werden müssen.
Ergänzen Sie `lib1` um die Dichtefunktion `phi(x)` wie oben und eine Funktion `Gauß(a, b)`, die das Integral $\int_a^b \varphi(x)\,dx$ mit Hilfe der (summierten) Simpson-Regel auf zehn Nachkommastellen genau berechnet. Benutzen Sie die Fehlerabschätzung der Vorlesung, um die nötige Anzahl der Teilintervalle zu bestimmen. (*Hinweis 1*: der Betrag der vierten Ableitung von $\varphi$ hat bei $0$ ein Maximum mit Wert $3/\sqrt{2\pi}$. *Hinweis 2*: Sie können außerdem Werte von $\varphi(x)$ für große $|x|$ vernachlässigen.) Berechnen Sie schließlich die neun Integrale für $a = 0$, $1$ und $2$ kombiniert mit $b = 1$, $4$ und $10^7$.

## Abgabe

- Laden Sie das Archiv `P1vorgabe.zip` von `moodle` herunter, entpacken Sie es, und testen Sie Ihre Programme, indem Sie `test1.py` im gleichen Verzeichnis mit `python` ausführen. Erhalten Sie `ERROR`, so entspricht Ihr Programm nicht der Spezifikation von oben. Erhalten Sie `FAIL`, so ist Ihr Programm zwar lauffähig, aber die berechneten Werte sind fehlerhaft.
- **Abgaben, bei denen der Test gar nicht durchläuft oder mit `ERROR`, werden nicht akzeptiert**, `FAIL` führt nur zu Punktabzug.
- Sie finden in obigem Archiv auch die Datei `info1.md` mit anzugebenden Informationem zu Ihrem Team und Ihrer Abgabe, **bitte füllen Sie diese nach dortiger Anleitung aus, und vergessen Sie nicht die Quellenangabe.** Abgaben mit unvollständiger Datei `info1.md` können nicht gewertet werden.
- Komprimieren und bündeln Sie alle oben erzeugten oder geänderten Dateien, indem Sie ein ZIP-Archiv erstellen. Sollten Sie nicht wissen, wie das geht, konsultieren Sie dazu die Dokumentation Ihres Betriebssystems.
- Benennen Sie Ihr ZIP-Archiv `P1.zip`.
- Schreiben Sie eine Email an `rosehr@hm.edu` mit Betreff `Numerik Abgabe`, Dateianhang `P1.zip` und irgendwelchem sonstigen Inhalt. **Der Automat akzeptiert die Abgabe nur, wenn diese Angaben (Betreff, Dateiname) korrekt sind.**

**Praktikumstermine P1:** keine; **Abgabetermin:** 04.05.