

# Aufgabenstellung Programmmentwurf

## Scientific Programming Lab

Gegeben ist eine Aufgabenstellung aus dem Bereich des Scientific Programming, die an die Lektionen der Vorlesung angelehnt ist. Zu erstellen ist ein Jupyter-Notebook analog zu denen in der Vorlesung verwendeten, welches die Lösungen aller Aufgabenstellungen sowie relevante Tests, Visualisierungen, Erläuterungen, etc. enthält. Die Bearbeitung erfolgt in Gruppen mit jeweils genau 5 Personen (Ausnahmen bei abweichender Kursgröße in Absprache mit dem Dozenten). Jede Gruppe bearbeitet und präsentiert die gestellte Aufgabe gemeinsam. Die Gruppeneinteilung erfolgt über das Moodle-Kurssystem.

### **Teil 1. Riemann Integrale mit Ober- und Untersummen, Trapezregel**

Erklären Sie die Methoden der numerischen Integration mittels Obersummen, Untersummen und der Trapezregel.

Entwerfen Sie eine Funktion, die eine numerische Integration durchführt.

Die Funktion soll eine aufrufbare Funktion (callable, siehe Notebook 7), sowie nötige Parameter für die Wahl des Integrationstyps und dessen Parametrierung übergeben bekommen. Achten Sie insbesondere auf Definitionslücken und Nulldurchgänge.

### **Teil 2. Integration mittels Monte-Carlo-Simulation**

Legen Sie die Grundlagen für die Integration mittels Monte-Carlo-Simulation dar. Erweitern Sie ihren Code, um diese durchführen zu können. Ermöglichen Sie sowohl eine manuelle Wahl des Sampling-Bereichs, als auch eine explorative Methode, die also obere und untere Grenzen selbst bestimmen muss.

### **Teil 3. Darstellung der Integrationsmethode**

Überlegen Sie sich, wie Sie die Ergebnisse aus Teil 1 und Teil 2 visualisieren können. Wählen Sie eine geeignete Schnittstelle, bzw. Integration der Funktionalität.

### **Teil 4. Test der numerischen Integration**

Testen Sie ihre Funktion an verschiedenen interessanten Klassen von Funktionen. Insbesondere sollten die Funktionen  $\sin(x)$  auf dem Intervall  $[0, 2\pi]$ ,  $x^2 - 4x + 2$  auf dem Intervall  $[-3, 3]$ , sowie  $e^x$  auf dem Intervall  $[0, 10]$  getestet werden. Stellen Sie dar, wie sich die numerische Integration bei feiner werdender Parametrierung dem tatsächlichen Wert annähert. Erweitern Sie hierzu geeignete zu integrierende Funktionen um eine Stammfunktion, die als Grundlage für den wahren Wert genutzt werden kann.

### **Teil 5. Grenzen numerischer Integration**

Führen Sie die numerische Integration der Funktion  $\sin((x + \frac{1}{32}) * 16\pi)$  auf dem Intervall  $[0, 1]$  mit Obersummen, Untersummen oder der Trapezregel durch. Vergleichen Sie die Ergebnisse bei der Aufteilung auf 2, 4, 8 und 16 gleichgroße Intervalle und erklären Sie die Ergebnisse.

Können Sie Kriterien formulieren, die gut Ergebnisse der numerischen Integration garantieren?

Optional: Zeigen Sie zum Beispiel anhand der Gaußschen Fehlerfunktion auf, dass es Funktionen gibt, die zwar nicht klassisch integrierbar sind, sich aber gut numerisch integrieren lassen.

## Bewertungskriterien und Abgabe

### 1. Mathematische Spezifikation (5 Punkte):

Beschreiben Sie (analog zu den Kapiteln der Vorlesung) die mathematischen Aspekte Ihrer Aufgabenstellung in einem Jupyter-Notebook. Verwenden Sie dazu die Vorlagen für Definition, Satz, etc. Geben Sie am Ende des Notebooks die von Ihnen verwendeten Quellen an.

### 2. Implementierung (10 Punkte):

Setzen Sie die für die Lösung der Aufgabenstellung notwendigen Programmkomponenten um (Klassen, Funktionen, etc.). Achten Sie auf Vollständigkeit, Korrektheit und Code-Dokumentation. Nutzen Sie Fehlerbehandlung, Exceptions, und aussagekräftige Fehlermeldungen, wo sie sinnvoll sind.

### 3. Test, Evaluation und Auswertung (10 Punkte):

Überprüfen Sie die Korrektheit Ihres Programms und beantworten Sie die gestellte Forschungsfrage der Aufgabestellung. Implementieren Sie dazu Tests und andere Evaluationen, die Ihre Lösung demonstrieren. Verwenden Sie, wo möglich, Visualisierungen wie in der Vorlesung gezeigt.

### 4. Technische Anforderung (5 Punkte):

Stellen Sie die Ablauffähigkeit in Jupyter sicher. Stellen Sie sicher, dass Sie sprechende Bezeichner und korrekte Terminologie verwenden. Stellen Sie sicher, dass Ihr Code nicht exzessiv lange läuft oder abstürzt. Geben Sie die Namen aller Gruppenmitglieder an.

**Abgabe** Bearbeitung in Gruppen mit jeweils **5-6 Personen** bis zum **Mittwoch, 16.07.2025 23:59 Uhr einzureichen über das Moodle Lernsystem**. Am **Donnerstag, 17.07.2025** werden die Programmentwürfe als Teil der Lehrveranstaltung im Kurs präsentiert.

**Abzugeben sind:**

1. **Programm:** Genau ein Jupyter-Notebook mit allen notwendigen Zusatzdateien. Diese muss durch Kopieren aller enthaltenen Dateien in das Startverzeichnis einer Jupyter-Instanz lauffähig sein. Die Installation zusätzlicher Module über die in der Vorlesung genutzt oder in Jupyter standardmäßig enthaltenen hinaus darf nicht notwendig sein.
2. **PDF** des Jupyter-Notebooks mit allen Ergebnissen (Diagramme, Code-Output, etc.).