

3. Aufgabenblatt

Reichen Sie die Lösungen zu den Aufgaben bitte bis Montag, den 22. Mai 2017 um 23⁵⁵ Uhr über ISIS ein. Als Abgabeformat verwenden Sie bitte wieder ein zip-komprimiertes tar-Archiv, in welchem Sie alle für die Lösung relevanten Dateien zusammenfassen. Vergessen Sie nicht das `makefile`!

Aufgabe 7

Schreiben Sie ein Programm zur Berechnung der Exponentialfunktion zur Basis e anhand ihrer Reihendarstellung. Der zu berechnende Funktionswert soll vom Benutzer eingegeben werden. Verwenden Sie für Ihr Programm die Vorlage `aufgabe07.cpp`, die Sie zusammen mit diesem Übungsblatt bei ISIS finden.

- ▶ Recherchieren Sie die Reihendarstellung für die Exponentialfunktion.
- ▶ Definieren Sie eine Funktion `fakultaet`, welche die Fakultät einer „beliebigen“ Zahl n mit $n \in \mathbb{N}_0$ berechnet und zurückgibt.
- ▶ Definieren Sie eine Funktion `power`, welche die ganzzahlige Potenz x^n einer beliebigen Zahl x mit $x \in \mathbb{R}$ berechnet ($n \in \mathbb{N}_0$) und zurückgibt. Verwenden Sie hierfür *nicht* die fertigen Funktionen aus einer Bibliothek (zum Beispiel `cmath` oder `math.h`).
- ▶ Definieren Sie eine Funktion `myExp` zur Berechnung von $\exp(x)$, deren optionaler Parameter die Anzahl der Reihenglieder enthält.
- ▶ Vergleich Sie das Ergebnis Ihrer Berechnung mit dem Ergebnis der in der Standardbibliothek enthaltenen `exp`-Funktion, indem Sie die Differenz ausgeben lassen.
- ▶ Experimentieren Sie mit dem Quellcode. Wo liegen die Grenzen Ihrer Berechnung und wie könnten sie erweitert werden? Schreiben Sie die Antwort zu dieser Frage in den Quellcode ihres Programms.

(6 Punkte)

Aufgabe 8

Schreiben Sie ein Programm, das die n . Primzahl berechnet, wobei n vom Benutzer eingegeben werden soll. Die Eingaben des Benutzers sollen bis zu einem geeigneten Abbruchkriterium wiederholt abgefragt werden.

- ▶ Recherchieren Sie zunächst einen Algorithmus, der eine Zahl darauf prüft, ob es sich um eine Primzahl handelt. Implementieren Sie anschließend eine Funktion `isPrime`, die einen ganzzahligen positiven Wert erwartet und einen booleschen Wert zurückgibt:

```
1  bool isPrime (unsigned int zahl)
2  {
3      ... // Algorithmus
4  }
```

- ▶ Implementieren Sie eine weitere Funktion `nprime`, die die n . Primzahl berechnet und hierfür `isPrime` aufruft.
- ▶ Lagern Sie die beiden primzahlspezifischen Funktionen in eine separate Datei aus, und erstellen Sie die entsprechende Header-Datei mit der Deklaration der extern definierten Funktionen, die Sie im Hauptprogramm einbinden.
- ▶ Im Hauptprogramm soll die Position der zu findenden Primzahl von der Standard-eingabe gelesen, die Funktion `nprime` aufgerufen und das Ergebnis ausgegeben werden.

(7 Punkte)

Hinweis

In Ihrem `tar`-Archiv sollten folgende Dateien enthalten sein:

- ▶ `aufgabe07.cpp` (Antwort nicht vergessen!)
- ▶ `aufgabe08.cpp`
- ▶ `prime.cpp` und `prime.hpp`
- ▶ `Makefile` (Beachten Sie die Abhängigkeiten!)