TU Berlin, Institut für Mathematik
Thomas Slawig, Uwe Prüfert, Henrik Büsing

6. Übungsblatt

Abgabe Theorie bis zum 13.12.05

`www.math.tu-berlin.de/Vorlesungen/WS05/ProgMa`

Werden in einer Funktion wiederholt die gleichen Anweisungen ausgeführt, um einen Wert zu berechnen, so nennt man das eine iterative Berechnung oder **Iteration**. Beispiel: Berechnung der Wurzel im 4. Tutorium.

Ruft sich eine Funktion selbst auf, um den Funktionswert zu berechnen, so nennt man das eine rekursive Berechnung oder **Rekursion**. Beispiel: Berechnung des Binomialkoeffizienten im 6. Tutorium mit der Formel

$$\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}.$$

Schreiben Sie zu den folgenden beiden Aufgaben jeweils Funktionen als Pseudocode, Struktogramm, oder Flussdiagramm. Benennen Sie dabei immer den Namen der Funktion, die Eingabeparameter und den Rückgabewert.

1. Aufgabe (5 Punkte)
Schreiben Sie eine iterative und eine rekursive Funktion zur Berechnung von x^n für $x \in \mathbb{R}, n \in \mathbb{N}$ (mit $0 \in \mathbb{N}$), ohne die Potenzfunktion zu benutzen. Geben Sie jeweils die Anzahl der benötigten Operationen an. Multiplikationen, Zuweisungen, Vergleiche und Funktionsaufrufe zählen jeweils als eine Operation. Zählen Sie die Operationen zur Ausführung einer *for*-Schleife mit, wenn Sie diese benutzen.

2. Aufgabe (3 Punkte)
Schreiben Sie eine rekursive Funktion zur Berechnung der Fibonaccizahlen (vgl. 5. Tutorium). Wieviele Funktionsaufrufe benötigt man in Abhängigkeit von n zur Berechnung des n-ten Folgengliedes?

5. Programmieraufgabe (Vorführen bis zum 13.12.05)
Es gibt eine explizite Lösungsformel für die Nullstellen eines kubischen Polynoms, vgl. etwa

`www.montgelas-gymnasium.de/mathe/kubfa/leitkubgleich.html`

Entwickeln Sie einen Algorithmus, der alle Nullstellen berechnet, und schreiben Sie eine C++-Funktion, die ihn implementiert.
Eingabe: Koeffizienten $a, b, c, d \in \mathbb{R}$ (ohne Einschränkung).
Ausgabe: Real- und Imaginärteile aller drei Nullstellen.
Alle Sonderfälle ($a, b, c = 0$) sollen behandelt werden. Benutzen Sie dazu eine Funktion zur Nullstellenberechnung eines quadratischen Polynoms, s. Tutorium.
Gibt es für bestimmte Parameterwerte weniger als drei Nullstellen, so soll für die anderen der Wert `nan` (z. B. als Ergebnis des Ausdruckes `0/0`) zurückgegeben werden.

Testen Sie die Funktion in einem Hauptprogramm.

Hinweis: Der Aufruf `pow(x,a)` der Funktion aus `cmath` berechnet für $a \notin \mathbb{Z}$ das richtige Ergebnis nur für $x \geq 0$. Wie kann man damit dennoch z.B. $\sqrt[3]{-x} = -\sqrt[3]{x}$ erhalten?

Es hat sich als Arbeitserleichterung bewährt, einen Algorithmus vor dem Beginn der Programmierung in einer Dokumentationstechnik aufzuschreiben.