

Übungen zu Einführung in die Informatik I

Aufgabe 4 **Lineare Rekursion**

Rekursion, auch Rekurrenz oder Rekursivität (von lateinisch recurrere = zurücklaufen) ist ein allgemeines Prinzip zur Lösung von Problemen. In vielen Fällen ist die Rekursion eine von mehreren möglichen Problemlösungsstrategien, sie führt oft zu eleganten mathematischen Lösungen. Als Rekursion bezeichnet man den Aufruf oder die Definition einer Funktion durch sich selbst.

- a) Implementieren Sie die Fakultätsfunktion

$$fak(n) = n!$$

- b) Implementieren Sie die Zahlensumme

$$sum(n) = \sum_{i=1}^{i=n} i$$

- c) Testen Sie Ihre Implementationen geeignet

Aufgabe 5 **Fibonacci Zahlen**

Die Fibonacci-Folge ist eine mathematische Folge von positiven ganzen Zahlen, den Fibonacci-Zahlen. Der Mathematiker Leonardo Fibonacci (Leonardo von Pisa) entwickelte sie, um das Wachstum einer Population von Kaninchen zu beschreiben, und publizierte sie in seinem Buch Liber Abaci aus dem Jahre 1202.

Die Folge ist rekursiv definiert durch:

•

$$fib(0) = 0$$

•

$$fib(1) = 1$$

•

$$fib(n) = fib(n-1) + fib(n-2)$$

Diese Rekursion gehört zur Klasse der Verzweigten Rekursion (fat rekursion). Jeder Aufruf der rekursiven Funktion löst direkt zwei oder mehr Aufrufe aus.

- a) Implementieren Sie die Fibonaccifunction

$$fib(x) = fib(x-1) + fib(x-2)$$

geeignet in ocaml

- b) Testen Sie die Laufzeit der Fibonaccifunktion mit verschiedenen Eingaben.

Aufgabe 6 Türme von Hanoi

Die Türme von Hanoi sind ein mathematisches Knobel- und Geduldsspiel. Das Spiel besteht aus drei Feldern, auf die Scheiben verschiedener Größe gelegt werden können. Zu Beginn sind alle Scheiben auf einem Feld, der Größe nach geordnet, mit der größten Scheibe unten und der kleinsten oben.

Bei jedem Zug darf die oberste Scheibe eines beliebigen Feldes auf eines der beiden anderen Felder gelegt werden, vorausgesetzt dort liegt nicht schon eine kleinere Scheibe. Ziel des Spiels ist es den kompletten Scheiben-Stapel auf ein anderes Feld zu versetzen.

- a) definieren Sie eine Hilfsfunktion, die die Position des Turms darstellt
- b) definieren Sie eine Hilfsfunktion, die einen Zug darstellt
- c) definieren Sie die Funktion, die den Turm von Hanoi verschiebt

Aufgabe 7 Babylonisches Wurzelziehen

Das Babylonische Wurzelziehen (oft auch Heron-Verfahren) ist ein alter iterativer Algorithmus zur Bestimmung einer rationalen Näherung der Quadratwurzel einer positiven Zahl a . Dabei wird folgende Formel verwendet:

$$x_{n+1} = (x_n + a/x_n) / 2$$

- a) Verwenden Sie diese rekursive Formel, um eine Funktion `babylon k a` zu definieren, die die Wurzel der positiven Zahl a durch k Iterationen approximiert. Brechen Sie dazu die Rekursion nach k Schritten ab. Als Startwert der Rekursion verwenden Sie $x_0 = 1$.
- b) Testen Sie Ihre Approximation der Wurzel mit einigen Werten für a und k . Finden Sie heraus ab welchem k die Genauigkeit ausreichend ist.
- c) Schätzen Sie die Anzahl der rekursiven Aufrufe in Abhängigkeit von k ? Zeichnen Sie die Aufrufstruktur für $k = 3$.
- d) Alternativ kann man die Wurzel einer positiven Zahl a (hier darf angenommen werden, dass $a > 1$ ist) auch durch Intervallschachtelung berechnen. Initial soll mit dem Intervall $[0, a]$ begonnen werden und in jedem Schritt soll das Intervall halbiert werden. Schreiben Sie eine Funktion `intervall k a (l, r)` die das Intervall $[l, r]$ mit $\sqrt{a} \in [l, r]$ nach k Iterationen liefert. Testen sie die Approximation mit einigen Werten.
- e) Schreiben Sie eine Funktion `intervall_f f a (l, r)` die mit der Iteration abbricht sobald der maximale Fehler $(r - l)$ kleiner als f ist und das gefundene Intervall zurückliefert.

Aufgabe 8 (H) Anzahl der rekursiven Aufrufe bei Fakultäts- und Fibonaccifunktion

Die Analyse der Funktionen, die auf diesem Übungsblatt zu erstellen waren, verdeutlichen, dass die Anzahl der rekursiven Aufrufe von der Struktur der Rekursion abhängt. Deshalb sollen in dieser Aufgabe Funktionen erstellt werden, mit deren Hilfe für die jeweilige Funktion diese Anzahl bestimmt werden soll.

- a) Geben Sie die Implementierung einer rekursiven Funktion an, die die Anzahl der Aufrufe der Fakultätsfunktion zählt. (Hinweis: Die gesuchte Funktion weist dieselbe Aufrufstruktur wie die Fakultätsfunktion auf, jedoch muss das Ergebnis der Funktion bei jedem rekursiven Aufruf um 1 erhöht werden)
- b) Geben Sie eine rekursive Funktion an, die die Anzahl der Aufrufe der Fibonacci-Funktion zählt.
- c) Wie lässt sich die Anzahl der Aufrufe in Abhängigkeit vom Funktionsparameter n abschätzen?

Aufgabe 9 (H) Frühester Ostertermin

Auf dem Aufgabenblatt 1 wurde ein Algorithmus zur Berechnung des Ostertermins implementiert.

- a) Der Algorithmus zur Berechnung des Ostertermins bedarf noch einer Ergänzung um folgende Zusatzregel:
 - Statt dem 26. April findet Ostern stets am 19. April statt
 - Statt dem 25. April findet Ostern am 18. April statt, wenn (siehe Blatt 1) $d = 28 \wedge e = 6 \wedge a > 10$.

Ergänzen Sie Ihre Implementierung um diese Zusatzregel.

- b) Erstellen Sie eine rekursive Funktion, die den frühesten Ostertermin ermittelt. (Hinweis: Die Typdefinition der Monate aus Aufgabenblatt 1 erlaubt auch Vergleiche zwischen diesen Monaten, d.h. `April < März = false`)