Dirk Praetorius
Michele Ruggeri

Sommersemester 2015
14.04.2015

Übungen zur Vorlesung
Einführung in das Programmieren für TM

Serie 4

Aufgabe 4.1. Write a *nonrecursive* function `binomial` which computes the binomial coefficient $\binom{n}{k}$. Use an appropriate loop and the identity $\binom{n}{k} = \frac{n\cdot(n-1)\cdots(n-k+1)}{1\cdot 2\cdots k} = \frac{n}{1}\cdot\frac{n-1}{2}\cdots\frac{n-k+1}{k}$. Additionally, write a main program that reads in the values $k, n \in \mathbb{N}_0$ with $k \leq n$ and prints out $\binom{n}{k}$. Save your source code as `binomial.c` into the directory `serie04`.

Aufgabe 4.2. The Fibonacci sequence is defined by $x_0 := 0$, $x_1 := 1$ and $x_{n+1} := x_n + x_{n-1}$. Write a *nonrecursive* function `fibonacci(k)`, which, given an index k, computes and returns x_k. Then, write a main program which reads k from the keyboard and displays x_k. Save your source code as `fibonacci.c` into the directory `serie04`.

Aufgabe 4.3. Compute the sum $\sum_{j=1}^{n}(-1)^j/j$. From a numerical point of view, it is attractive to first sum the negative and positive contributions separately and then return the sum of these partial sums. Why could this be advantageous? Write a function `sum` which realizes this. Furthermore, write a main programm which reads in $n \in \mathbb{N}$ and prints out $\sum_{j=1}^{n}(-1)^j/j$. Save your source code as `sum.c` into the directory `serie04`.

Aufgabe 4.4. Write a function `geometricMean` that computes and returns the geometric mean value

$$\overline{x}_{\text{geom}} = \sqrt[n]{\prod_{j=1}^{n} x_j}$$

of a given vector $x \in \mathbb{R}^n_{\geq 0}$ Furthermore, write a main program that reads in $x \in \mathbb{R}^n$ and computes the geometric mean value thereof. The length $n \in \mathbb{N}$ of the vector should be a constant in the main program, but the function `geometricMean` should be implemented for arbitrary lengths. Save your source code as `geometricMean.c` into the directory `serie04`.

Aufgabe 4.5. Write a function `scalarproduct`, which, given two vectors $x, y \in \mathbb{R}^n$, computes the scalar product $x \cdot y := \sum_{j=1}^{n} x_j y_j$. Then, write a main program which reads the vectors x and y from the keyboard and displays $x \cdot y$. The length n of the vector should be constant in the main program, but the function `scalarproduct` should be programmed for arbitrary lengths n. Save your source code as `scalarproduct.c` into the directory `serie04`.

Aufgabe 4.6. Write a `void`-function `multiple(k,nmax)`, which computes and displays all the integer multiples of $k \in \mathbb{N}$ which are $\leq n_{\max} \in \mathbb{N}$. For instance, for $k = 5$ and $n_{\max} = 19$, the function yields the output

```
1 x 5 = 5
2 x 5 = 10
3 x 5 = 15.
```

Then, write a main program, which reads the values k and n from the keyboard and calls `multiple(k,nmax)`. Save your source code as `multiple.c` into the directory `serie04`.

Aufgabe 4.7. Write a function `double powN(double x, int n)` which computes x^n for all exponents $n \in \mathbb{Z}$ and $x \in \mathbb{R}$. It holds $x^0 = 1$ for all $x \in \mathbb{R}$. For $n < 0$ use $x^n = (1/x)^{-n}$. Moreover, $0^n = 0$ for $n > 0$. The term 0^n for $n < 0$ is not defined. In that case, the function should return the value `0.0/0.0`. You must not use the function `pow` from the math library. Save your source code as `powN.c` into the directory `serie04`.

Aufgabe 4.8. Explain the term *recursion*. Which components does a recursive function consist of?