

Scientific programming in mathematics

Exercise sheet 2

IF statement, Functions, Recursion, Vectors

Exercise 2.1. Write a function `evenorodd` which takes a number $n \in \mathbb{N}$ as input and returns the value 1 if n is even or 0 if n is odd. Write a main program which reads in the value n from the keyboard and prints on the monitor if n is even or odd.

Exercise 2.2. Write a function `rounding`, which, given $x \in \mathbb{R}$, computes the number $n \in \mathbb{Z}$ which is closest to x . If x is exactly in the middle between two integers n and $n + 1$, the function chooses the biggest one, i.e., $n + 1$. Then, write a main program which reads the number x from the keyboard, calls the function and displays the rounded value. Save your source code as `rounding.c` into the directory `series02`.

Exercise 2.3. Write a void-function `divisor` which checks if a given number $x \in \mathbb{N} := \{1, 2, 3, \dots\}$ is divisible by 2, 3, or 6. Additionally, write a main program that reads in the number x , then calls the function `divisor`, and prints out the result. Save your source code as `teiler.c` into the directory `series02`.

Exercise 2.4. Write a program that reads from the keyboard three numbers $x, y, z \in \mathbb{R}$ and prints them on the screen in increasing order, i.e., first the minimum $\min\{x, y, z\}$ and last the maximum $\max\{x, y, z\}$. Save your source code as `sort3.c` into the directory `series02`.

Exercise 2.5. Write a recursive function `binomial` which computes and returns the binomial coefficient $\binom{n}{k}$ of two given integers $0 \leq k \leq n$. Use the addition formula

$$\binom{n}{k} = \binom{n-1}{k} + \binom{n-1}{k-1} \quad \text{for } 1 \leq k < n$$

with $\binom{n}{0} = 1 = \binom{n}{n}$. Write a main program which reads k and n from the keyboard and prints to the screen the result $\binom{n}{k}$. Save your source code as `binomial.c` into the directory `series02`.

Exercise 2.6. The Fibonacci sequence is recursively defined by $x_0 := 0$, $x_1 := 1$, and $x_{n+1} := x_n + x_{n-1}$. Write a recursive function `fibonacci`, which computes and returns x_n for given $n \in \mathbb{N}_0$. Then, write a main program, which reads n from the keyboard and prints to the screen the value of x_n . Save your source code as `fibonacci.c` into the directory `series02`.

Exercise 2.7. Write a function `scalarproduct` which computes and returns the scalar product $w = \mathbf{u} \cdot \mathbf{v} := ax + by + cz$ of two given vectors $\mathbf{u} = (a, b, c)^T$ and $\mathbf{v} = (x, y, z)^T$. Furthermore, write a main program which reads the parameters a, b, c, x, y, z from the keyboard and prints to the screen the value of the scalar product. Save your source code as `scalarproduct.c` into the directory `series02`.

Exercise 2.8. Write a program that allocates a static vector x of length 1000. The coefficients shall satisfy $x[i] = i$ for all $i \in \{0, 1, \dots, 999\}$. Next, the vector shall be displayed on the screen. You must not use loops. Save your source code as `array.c` into the directory `series02`.

Hint: Write functions `createVector` and `printVector` that are called in the main program.