

Programming Techniques for Scientific Simulations

Exercise 0

Problem 0.1 Recursion and loop: greatest common divisor

Write a program which reads in two positive integers a , b and computes their greatest common divisor $\gcd(a, b)$. Use Euclid's algorithm based on relation

$$\gcd(a, b) = \gcd(a - b, b) = \dots = \gcd(a \bmod b, b). \quad (1)$$

First implement the algorithm by recursive function call and afterwards in a non-recursive way using a loop.

Problem 0.2 Dynamic programming: Fibonacci sequence

Write a program which reads in a non-negative integer n and prints out the n -th number of the Fibonacci sequence, F_n . The sequence is defined by

$$F_n = \begin{cases} 0 & n = 0, \\ 1 & n = 1, \\ F_{n-1} + F_{n-2} & n > 1. \end{cases} \quad (2)$$

What is the value of F_{82} ?

Problem 0.3 Fibonacci sequence revisited

The recurrent relation for the Fibonacci sequence may be rewritten in a matrix form,

$$\begin{pmatrix} F_{n-1} \\ F_n \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} F_{n-2} \\ F_{n-1} \end{pmatrix} \quad (n > 1). \quad (3)$$

Can you use it to make the computation of F_n faster?

Problem 0.4 Divide and conquer: merge sort

Write a program which reads in a positive integer n , generates a non-monotonic sequence of n numbers¹ and sorts it using merge sort. Merge sort is an algorithm of the divide-and-conquer class of algorithms – a sequence is sorted according to these rules:

- If $n == 1$, then the sequence is sorted.
- For $n > 1$ we split the sequence to left and right half and apply sort on each of the halves separately. Afterwards we merge the separately sorted halves to obtain the original sequence sorted as a whole.

¹The sequence may be generated by $\sin(i)$ for $i = 0, 1, \dots, n - 1$.