

Scientific Programming (Wissenschaftliches Programmieren)

Exercise 3

Fibonacci numbers

- Create a Python-script / IPython-notebook which generates Fibonacci numbers.
- The first two elements of the Fibonacci series are 1, otherwise each element is the sum of the previous two elements.
- The script should read the number of desired Fibonacci terms from standard input.
- It should first create a list with all requested Fibonacci-numbers and then print the content of the list, one Fibonacci-number per line.
- The script should make sure that the numbers are aligned correctly. The field width should be determined based on the last Fibonacci-number calculated.
- Make sure the script also works correctly when the desired number of Fibonacci terms is only one or two.

Alphabetical word list

- Create a Python-script / IPython-notebook which lists all words occurring in a text.
- The script should read a sentence (words separated by space) from standard input.
- It should create a list with all the occurring words in alphabetical order and print the content of this list in one line. The words in the output should be separated by semi-colon (e.g. word1; word2; word3).
- Every word should be printed only once, even if it occurs multiple times in the input.
- The words in the input should be converted to lower case before being processed, so that differences in upper and lower case are ignored.

Word occurrence

- Create a Python-script / IPython-notebook which prints a statistics about which words and how often do occur in a text.
- The script should read a sentence (words separated by space) from standard input.
- It should print the list of all words occurring in the input and the number of their occurrence.
- It should print the most abundant word first and the least abundant last (ordered descendingly by occurrence)
- The words in the input should be converted to lower case before being processed, so that differences in upper and lower case are ignored.