

Exercises for Practical 2

In this practical you will work on writing Python functions and scripts in an IDE. Use a new file for each question.

1. Write a script to compute the coefficients of the degree n power series expansion of $\cos(x)$. What is the sum of the first 15 coefficients? (Include the zero coefficients in the sum.)
2. Write a Python function that returns the n th Fibonacci number. Write a test that only runs when the file is called directly that checks the 17th number.
3. Using Newton's method and starting from the initial guess $x_0 = 0$, find a root of $\sin(x) + \cos(x)$. Then, find another root by starting from the initial guess $x_0 = 6$.
4. How many prime numbers in the range $[10, 100]$ are 'mirror primes' (in the sense that if a two-digit number mn is prime, its reverse nm is also prime)?
5. Find the sum of all the multiples of 3 or 5 below 1000.
6. What is the smallest positive number that is exactly divisible by all of the numbers from 1 to 10?
7. Implement a bisection algorithm to find the root of a given function $f(x)$ on a given interval $[a, b]$. Your function should take f , a , and b as inputs.
8. Python has a built-in function `bin()` that converts an integer to its binary representation as a string (e.g., `bin(10)` returns `'0b1010'`). Write your own function, `my_dec_to_bin()` in a new file, which takes an integer and returns a string of only the binary digits (e.g., for an input of 10, it should return `'1010'`).
9. Given a function, an interval $[a, b]$, and a parameter N , write your own `trap_integral` in a new file to compute the N -point trapezoidal rule approximation of the integral of the input function. (Hint: Check Wikipedia for the trapezoidal rule formula.)
10. Write a script which imports the functions from questions 7, 8, and 9 and runs them with various inputs.