PH504M Lab 4: Recap of basic syntax

Vikram Khaire

31 Januray 2025

Questions

1. **Print a Message**

   Write a function `greet()` that prints `"Hello, class!"`.

2. **Print Even Numbers**

   Write a function `print_evens(n: int)` that prints the first n even numbers (each on a new line).

3. **Compute Mean**

   Write a function `compute_mean(data: list) -> float` that returns the mean of a list of numbers.

4. **Compute Standard Deviation**

   Write a function `compute_std(data: list) -> float` that returns the standard deviation of a list of numbers.

5. **Sum of First N Natural Numbers**

   Write a function `sum_natural(n: int) -> int` that returns the sum of the first n natural numbers.

6. **Check Even or Odd**

   Write a function `is_even(n: int) -> bool` that returns `True` if n is even, otherwise `False`.

7. **Factorial of a Number**

   Write a function `factorial(n: int) -> int` that returns the factorial of n.

8. **Reverse a list**

   Write a function `reverse(s: list) -> list` that returns the reverse of the given string.

9. **Generate a List of Squares**

   Write a function `squares(n: int) -> list` that returns a list of squares of numbers from 1 to n.

10. **Find the Maximum and Minumum out of three number list**

    Write a function `find_max(data: list) -> int` that returns the maximum and minimum number in a given list of three numbers.

11. **Sum of Digits of a Number**

    Write a function `sum_digits(n: int) -> int` that returns the sum of the digits of n.

12. **Simple Line Plot**

    Write a function `plot_line(x: list, y: list) -> None` that takes two lists of numbers and plots them using Matplotlib.

13. **Fibonacci Sequence**

    The Fibonacci sequence is defined as follows:

    $$\begin{aligned} &F_0 = 0, \\ &F_1 = 1, \\ &F_n = F_{n-1} + F_{n-2} \quad \text{for } n \geq 2. \end{aligned}$$

    Write a function `fibonacci(n: int) -> list` that returns the first n Fibonacci numbers as a list.

14. **Roots of a Quadratic Equation**

    Write a function `quadratic_roots(a: float, b: float, c: float) -> tuple` that returns the roots of the quadratic equation:

    $$ax^2 + bx + c = 0$$

    using the quadratic formula.

15. **Count Words in a Sentence** Write a function `count_words(sentence: str) -> int` that returns the number of words in a given sentence.

16. **Check Armstrong Number**

    A number is an Armstrong number if the sum of its digits each raised to the power of the number of digits equals the number itself. For example, 153 is an Armstrong number because:

    $$1^3 + 5^3 + 3^3 = 153.$$

    Write a function `is_armstrong(n: int) -> bool` that checks if a number is an Armstrong number.

17. **Generate a Sine Wave Plot**

    Write a function `plot_sine_wave(frequency: float, duration: float)` that plots a sine wave with a given frequency and duration using Matplotlib.

18. **Convert Decimal to Binary**

    Write a function `decimal_to_binary(n: int) -> str` that converts a decimal number to its binary representation.

19. **Calculate Compound Interest**

    Write a function `compound_interest(p: float, r: float, t: int) -> float` that calculates the compound interest using the formula:

    $$A = P\left(1 + \frac{r}{100}\right)^t$$

    where P is the principal amount, r is the annual interest rate, and t is the time in years.

20. **Reverse a Number**

    Write a function `reverse_number(n: int) -> int` that returns the reverse of a given number.

Note

Students can use the `//` and `%` operators in some of these questions. Here's what they do:

- The `//` operator (floor division) returns the quotient of a division without the remainder. For example:

  $$10//3 = 3$$

- The `%` operator (modulus) returns the remainder of a division. For example:

  $$10\%3 = 1$$