

LABORATORIUM 9

Zestaw zadań do samodzielnego rozwiązania o różnym stopniu trudności, obejmujących języki Python i F# oraz różne paradygmaty programowania.

Python

Programowanie Proceduralne

1. Napisz program, który oblicza sumę cyfr liczby podanej przez użytkownika.
2. Utwórz kalkulator z podstawowymi operacjami matematycznymi (dodawanie, odejmowanie, mnożenie, dzielenie).
3. Utwórz program do rozwiązywania równania kwadratowego przy pomocy formuły kwadratowej.
4. Napisz skrypt do analizy danych z pliku CSV, który oblicza średnią dla wybranej kolumny.

Programowanie Obiektowe:

5. Stwórz klasę Prostokąt z polami długość i szerokość. Dodaj metody do obliczania pola i obwodu.
6. Zaimplementuj klasę Samochód, która dziedziczy po klasie Pojazd, i dodaj metodę jazda.

Programowanie Funkcyjne:

7. Używając funkcji wbudowanych (np. map, filter, reduce), znajdź wszystkie liczby parzyste w liście liczb oraz ich sumę.
8. Napisz funkcję rekurencyjną, która oblicza n-ty wyraz ciągu Fibonacciego.
9. Napisz program obliczający permutacje listy liczb za pomocą funkcji rekurencyjnej.
10. Zaimplementuj funkcję, która sprawdza, czy podana lista liczb jest palindromem, używając jedynie funkcji wbudowanych.

F#

Programowanie Proceduralne:

1. Napisz funkcję, która oblicza sumę liczb od 1 do n.
2. Utwórz program sprawdzający, czy liczba jest liczbą pierwszą.
3. Utwórz program, który wczytuje dane o uczniach i ich ocenach, a następnie generuje raport z ocenami.
4. Napisz funkcję, która sortuje listę liczb rosnąco bez użycia wbudowanej funkcji sortującej.
5. Opracuj program do analizy plików tekstowych, który zlicza liczbę wystąpień poszczególnych słów i generuje raport.

Programowanie Funkcyjne:

6. Zaimplementuj rekurencyjną funkcję obliczającą silnię liczby.
7. Używając funkcji List.map, zamień wszystkie litery w liście słów na wielkie litery.
8. Zaimplementuj algorytm znajdowania największego wspólnego dzielnika (NWD) dwóch liczb.
9. Stwórz program, który przekształca listę łańcuchów na listę ich długości, używając funkcji funkcyjnych.
10. Zaimplementuj analizator wyrażeń matematycznych, który obsługuje nawiasy i różne operatory.
11. Napisz funkcję generującą wszystkie możliwe podzbiory z listy elementów (power set).