Laboratorium 1

Temat: Wprowadzenie do oprogramowania Matlab oraz Pycharm dla języka Python

Celem zajęć jest zapoznanie się ze strukturą, podstawowymi operacjami i funkcjami używanymi w środowisku Matlab oraz Pycharm.

Matlab

Matlab jest potężnym narzędziem do obliczeń numerycznych i analizy danych, oferującym wiele operacji i funkcji. Oto kilka podstawowych operacji i funkcji w Matlab:

Obliczenia matematyczne: Matlab oferuje szeroki zakres operacji matematycznych, takich jak dodawanie, odejmowanie, mnożenie, dzielenie, potęgowanie, operacje macierzowe i wektorowe.

Tworzenie i manipulacja macierzami: Można tworzyć macierze i wektory oraz wykonywać na nich różnorodne operacje, takie jak transpozycja, odwracanie macierzy, indeksowanie, wycinanie fragmentów danych itp.

Plotowanie i wizualizacja danych: Matlab posiada funkcje do tworzenia wykresów 2D i 3D, histogramów, powierzchni, heatmap, wykresów punktowych i innych typów wizualizacji danych.

Algorytmy numeryczne: Oferuje wiele wbudowanych algorytmów numerycznych do rozwiązywania równań różniczkowych, optymalizacji, interpolacji, aproksymacji, analizy Fouriera, transformacji oraz wielu innych.

Przetwarzanie sygnałów i obrazów: Matlab dostarcza narzędzia do przetwarzania sygnałów audio, analizy obrazów, filtracji, detekcji krawędzi, segmentacji obrazów i innych operacji na danych sygnałowych oraz obrazach.

Analiza statystyczna: Oferuje zestaw funkcji statystycznych do wykonywania operacji takich jak obliczanie średniej, odchylenia standardowego, testów statystycznych, regresji, analizy wariancji (ANOVA) i innych.

Programowanie: Matlab umożliwia pisanie skryptów i funkcji, co pozwala na tworzenie własnych algorytmów, procedur i narzędzi do specyficznych zastosowań.

Interakcja z innymi językami i narzędziami: Możliwa jest integracja z językiem C/C++, Java, Pythonem oraz innymi narzędziami i pakietami obliczeniowymi.

Matlab oferuje bogaty zestaw funkcji i narzędzi do analizy danych, eksploracji, obliczeń numerycznych oraz wizualizacji, co czyni go popularnym narzędziem w dziedzinach naukowych, inżynieryjnych i innych dziedzinach wymagających analizy danych.

Pycharm

PyCharm, jest potężnym środowiskiem programistycznym dla Pythona, zawiera wiele podstawowych operacji i funkcji, które można wykonywać. Oto kilka z nich:

Tworzenie projektu: Można tworzyć nowe projekty, wybierając odpowiedni typ projektu Pythona, definiując interpreter i środowisko wirtualne.

Edycja kodu: PyCharm oferuje intuicyjny edytor z funkcjami takimi jak podpowiedzi kodu, sprawdzanie składni, automatyczne uzupełnianie, refaktoryzacja kodu, funkcje refactoringu (zmiany nazw, ekstrakcja metod itp.) i wiele innych.

Debugowanie: Można debugować swój kod w PyCharm, ustawiając punkty przerwania, obserwując wartości zmiennych w trakcie działania programu, analizując stos wywołań i inne.

Zarządzanie projektem: PyCharm umożliwia zarządzanie plikami i katalogami w projekcie, wyszukiwanie, przenoszenie, kopiowanie i inne operacje na plikach.

Integracja z systemami kontroli wersji: Obsługuje systemy kontroli wersji takie jak Git, co umożliwia łatwe zarządzanie kodem źródłowym i współpracę zespołową.

Testowanie: Można tworzyć testy jednostkowe, uruchamiać je, analizować ich wyniki i śledzić pokrycie kodu testami.

Zarządzanie bibliotekami: PyCharm pozwala zarządzać bibliotekami Pythona za pomocą menedżera pakietów (pip) bezpośrednio z interfejsu użytkownika.

Środowisko wirtualne: Umożliwia tworzenie i zarządzanie środowiskami wirtualnymi, co pomaga w izolacji zależności projektu.

Analiza kodu: Oferuje narzędzia do statycznej analizy kodu, takie jak sprawdzanie jakości kodu, znajdowanie błędów, podkreślanie nieoptymalnego kodu i inne.

Refaktoryzacja: Można ulepszać strukturę swojego kodu poprzez różne operacje refaktoryzacji, takie jak zmiana nazw, ekstrakcja metod, przekształcanie kodu, i wiele innych.

PyCharm oferuje również wiele innych funkcji, takich jak wsparcie dla różnych frameworków (Django, Flask, itp.), integrację z bazami danych, analizę wydajności i wiele innych, które pomagają w szybkim i efektywnym tworzeniu oprogramowania w języku Python.

Przykładowe zadania

Zadanie 1
Napisać prostą aplikację, wykorzystując GUI, która umożliwi poprzez klikanie na przycisku, losową zmianę koloru przycisku, lub formatki.

Zadanie 2
Napisać aplikację wykorzystującą obiekty typu “Edit Field (Numeric)” oraz obiekt typu “Axes”. Z obiektów typu “Edit” należy pobrać wartości będące minimalnym i maksymalnym zakresem argumentów oraz krokiem przesunięcia, a następnie wyliczyć wartości funkcji sinus dla wszystkich argumentów z zakresu. Wynik w postaci graficznej należy wyświetlić w obiekcie typu “Axes

Zadanie 3
Napisać prosty kod w Python obliczający sumę liczb od 1 do 10. Wynik należy wyświetlić w konsoli.

Zadanie 4
Napisać prosty kod w Python który definiuje funkcję sprawdzającą parzystość. Funkcja ta przyjmuje liczbę jako argument i zwraca odpowiednią wiadomość informującą, czy liczba jest parzysta czy nieparzysta. Następnie sprawdza się parzystość liczby ustawionej w kodzie i wyświetla wynik na ekranie.

Projekt finansowany w ramach programu Ministra Nauki i Szkolnictwa Wyższego pod nazwą „Regionalna Inicjatywa Doskonałości" w latach 2019 - 2023 nr projektu 020/RID/2018/19 kwota finansowania 12 000 000 PLN