

## **Informatyka geodezyjna II**

### **Projekt 1**

---

### **Transformacje**

---

Michał Bielecki    319294    Michał Chwałek    319305

#### **Grupa 1**

Zajęcia:

poniedziałek 12:15-14:00

Rok akademicki:

2022/23, Semestr 4

---

**Prowadzacy:** mgr inż. Andrzej Szeszko

---

# 1 Cel ćwiczenia

Celem ćwiczenia jest napisanie programu umożliwiającego wykonanie następujących transformacji współrzędnych:

- geocentryczne  $\rightarrow$  geodezyjne
- geodezyjne  $\rightarrow$  geocentryczne
- geocentryczne  $\rightarrow$  topocentryczne
- geodezyjne  $\rightarrow$  układ PL-2000
- geodezyjne  $\rightarrow$  układ PL-1992

W przypadku współrzędnych płaskich prostokątnych istnieje możliwość wyboru, jednej z trzech elipsoid odniesienie: **GRS80**, **WGS84**, **Krasowskiego**

## 2 Specyfikacja

Skrypt powstał z wykorzystaniem języka programowania **python** w wersji: **3.11.3** przy pomocy bibliotek: **numpy**, **math**, **argparse**, obsługiwany w systemie oprogramowania **Windows**. Program przyjmuje i zwraca dane w formie pliku z rozszerzeniem **.txt**

## 3 Przebieg ćwiczenia

Do stworzenia programu wykorzystano funkcje wykonujące transformacje stworzone na potrzeby Geodezji Wyższej I w semestrze 3. Wszystkie funkcje zostały użyte jako metody pod klasą: Transformacje w celu umożliwienia sprawnego przeliczania współrzędnych z wykorzystaniem różnych modeli elipsoid. W tym celu wykorzystano metodę **init**, a zmienne zależne od elipsoid zapisano z wykorzystaniem odwołania **self**, tak aby metody mogły "pobrać" parametry elipsoidy wybranej przez użytkownika. W celu zwiększenia czytelności kodu zastosowano wyrażenie warunkowe: **if name == "main":**. W przypadku wprowadzenia nieprawidłowego modelu elipsoidy program wysłał informację zwrotną z popełnionym błędem.

Podczas tworzenia programu napotkaliśmy problem z odczytem współrzędnych wielu punktów z pliku. W przypadku gdy użytkownik na końcu pliku ze współrzędnymi wejściowymi utworzył zbyt dużo pustych linii otrzymywaliśmy błąd zamiany zmiennej typu **string** na **float**. Problem ten udało nam się rozwiązać stosując klauzulę **try...except**. Dzięki zastosowaniu tej klauzuli udało się również uniknąć błędnego działania programu w przypadku gdy użytkownik utworzy puste linie w dowolnym miejscu w pliku ze współrzędnymi wejściowymi.