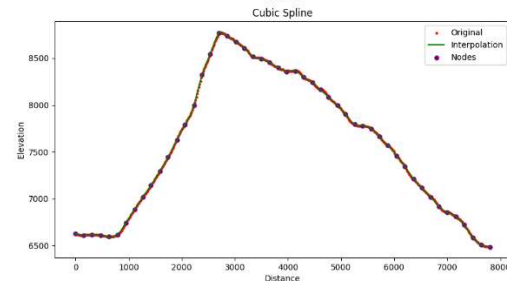
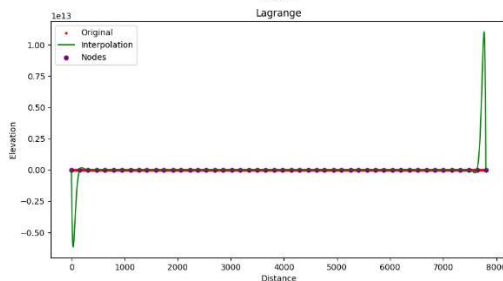
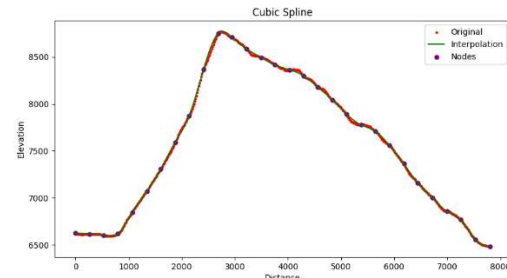
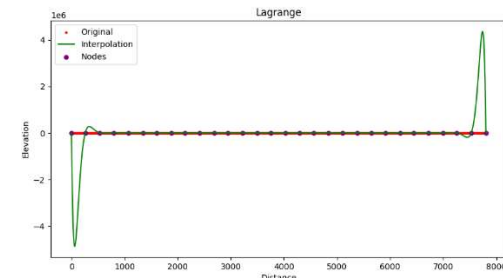
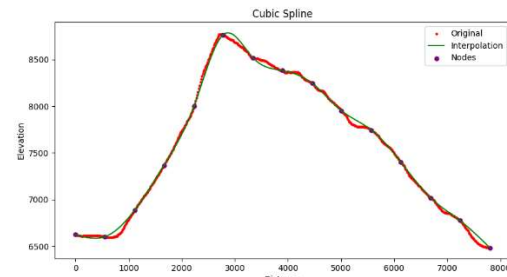
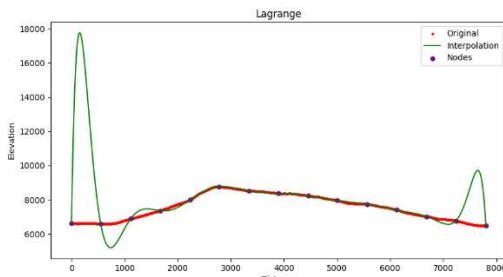
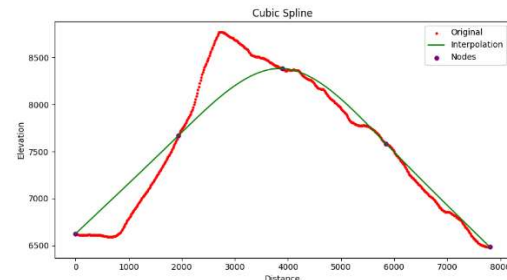
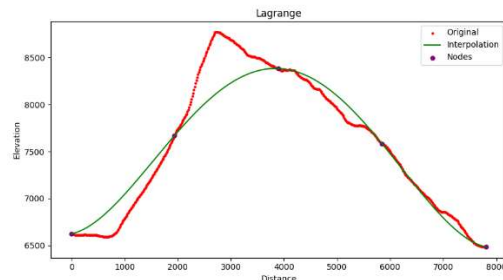


Sprawozdanie do projektu 3 – Interpolacja

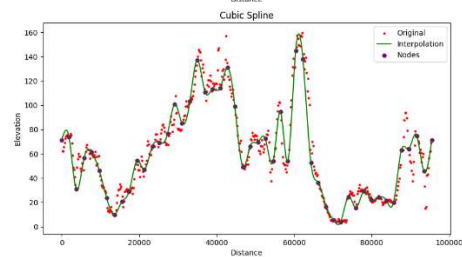
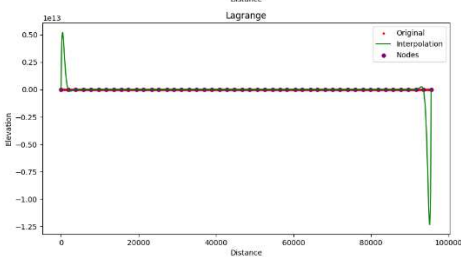
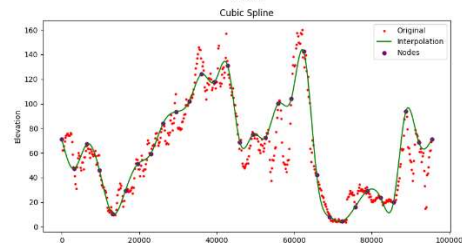
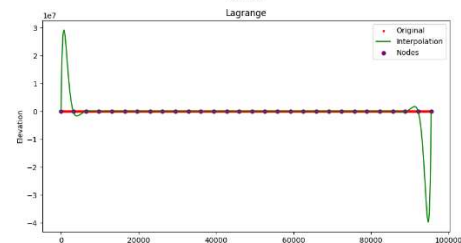
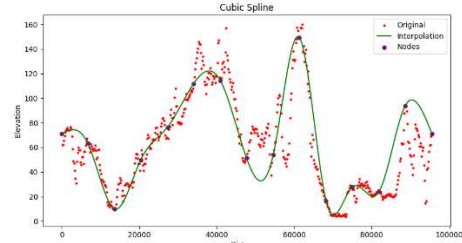
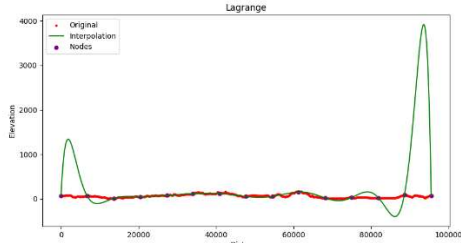
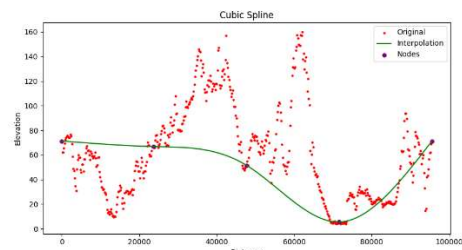
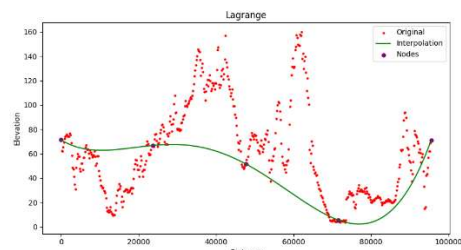
Adrian Ściepura 193350

- Wstęp teoretyczny
 - Celem projektu było zaimplementowanie dwóch metod interpolacji w celu stworzenia profilu wysokościowego kilku tras.
 - Metoda Lagrange'a
 - Jest to metoda która znajduje wielomian przechodzący przez dane punkty. Wielomian ten wyznaczany jest poprzez sumę funkcji bazowych. Gdzie funkcja bazowa jest iloczynem ilorazów różnic kolejnych wyrazów.
 - $\varphi_i(x) = \prod_{j=1, j \neq i}^{n+1} \frac{(x - x_j)}{(x_i - x_j)}$
 - $F(x) = \sum_{i=1}^n y_i \varphi_i(x)$
 - Metoda splejnow kubicznych
 - Metoda ta w przeciwieństwie do poprzedniej, skleja funkcje niskiego stopnia stworzone pomiędzy sąsiednimi węzłami. Tego typu podejście zapewnia gładkie przejście pomiędzy punktami.
 - $S_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3$
- Założenia projektowe
 - Język implementacji: Python
 - Zostały utworzone dwie klasy
 - Interpolation – klasa która przyjmuje ścieżkę do pliku .csv, a następnie go interpretuje i za pomocą funkcji show_plot rysuje wykresy profilu wysokościowego (z wykorzystaniem obu metod interpolacji)
 - InterpolationUtils – klasa która posiada wyłącznie metody statyczne – realizuje ona całą logikę obliczeń interpolacji
- Działanie
 - Mount Everest (Jedno znaczne wzniesienie)
 - Obserwacje
 - Dużo danych wejściowych (wysokie zagęszczenie punktów)

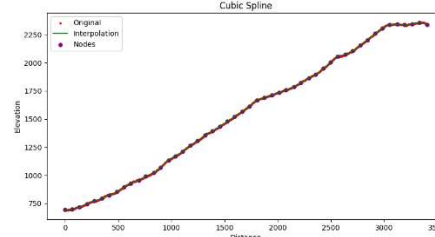
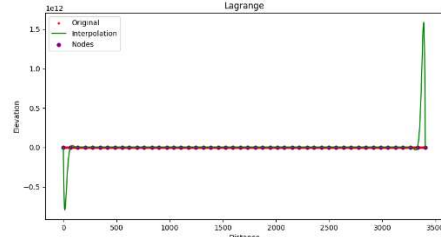
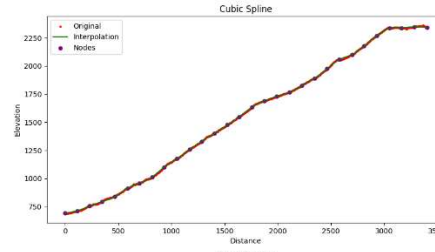
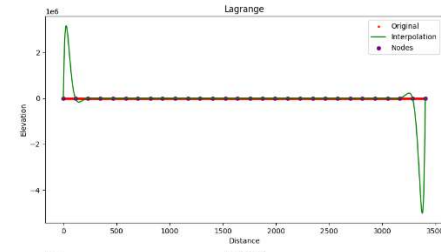
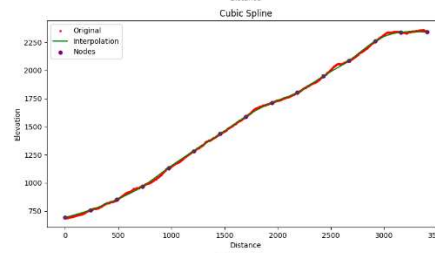
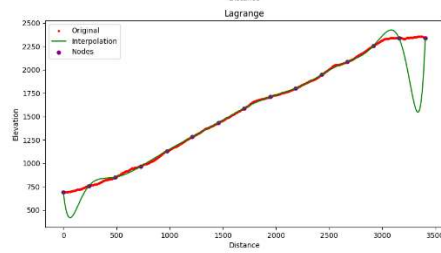
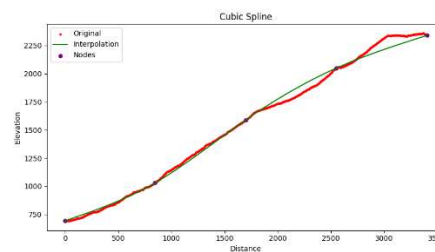
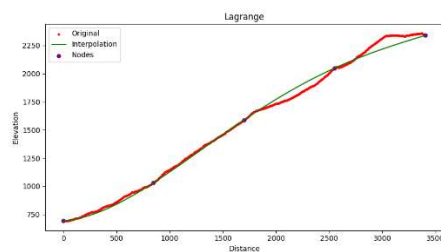
- Metoda lagrange -> początkowo wynik zbliżony do metody sklejania, po zwiększeniu liczby węzłów wynik staje się nieczytelny przez efekt rungiego (widoczny na krańcach)
- Metoda sklejania -> metoda działa bardzo dobrze w tym przypadku, po zwiększeniu liczby węzłów oba wykresy w znacznym stopniu się pokrywają



- Unsyncable_ride (Wiele wzniesień)
- Obserwacje
 - Mało pomiarów (niskie zagęszczenie danych wejściowych)
 - Metoda lagrange -> dla małej ilości węzłów – słabe dopasowanie, po zwiększeniu liczby węzłów – ponownie efekt rungiego powoduje że wynik jest nieczytelny
 - Metoda sklejania -> przy małej liczbie węzłów – efekt dokładnie ten sam co dla lagrange. Po zwiększeniu ilości węzłów dopasowanie staje się coraz dokładniejsze



- Stałe (liniowa zmiana wysokości)
- Obserwacje
 - Dużo danych wejściowych (wysokie zagęszczenie punktów)
 - Metoda Lagrange – nawet przy małej ilości punktów wykresy pokrywają się (funkcja stale rośnie – brak drastycznych zmian w jej przebiegu) – efekt rungiego jest trochę mniejszy niż w poprzednich przypadkach (nadal występuje ale na mniejszą skalę przez co wynik na drugim wykresie jest czytelny)
 - Metoda sklejania – wykresy bardzo szybko się pokrywają – już przy 15 węzłach pokrycie jest na tyle dobre że nie trzeba zwiększać ich ilości.



- Podsumowanie
 - Metoda Lagrange
 - Przy niskiej ilości punktów metoda radzi sobie całkiem dobrze – po zwiększeniu efekt rungiego zaburza wykres w znaczącym stopniu co utrudnia jego odczytanie. W przypadku użycia danych tej interpolacji do dalszej analizy błędy te mogłyby tragicznie wpłynąć na obliczenia
 - Metoda sklejania
 - Metoda bardzo efektywna – w każdym testowanym przypadku pokrycie było bardzo dokładne oraz nie zostały wykryte żadne problemy. Ze względu na to że metoda używa funkcji 3 stopnia wykresy mają bardzo płynne przejścia pomiędzy węzłami.
 - Wnioski
 - W rzeczywistości pisząc program nie wiemy jak będą wyglądać dane (mogą posiadać wiele uskoków) z tego powodu, aby zapewnić stabilność oraz niezawodność lepszą metodą wydaje się metoda sklejanie, która nie wykazała problemów z interpolacją testowanych (różnorodnych) danych