**Algorytmy Geometryczne**

**Lokalizacja punktu w przestrzeni dwuwymiarowej**

**Metoda trapezowa**

Oskar Blajsz

Maciej Wiśniewski

Grupa 3 Poniedziałek 16.45 A

Data wykonania 04.01.2025 Data oddania 05.01.2025

1. **Wprowadzenie**

W ramach projektu przygotowano i zaimplementowano struktury oraz bazujący na nich algorytm, którego celem jest realizacja i wizualizacja **trapezowej** metody lokalizacji punktu na przestrzeni dwuwymiarowej. Początkowym etapem algorytmu jest obszar z **podziałem poligonowym**, wcześniej przygotowany tudzież zapodany przez użytkownika. Podany jest też dowolny **punkt P**, należący do wspomnianego poprzednio obszaru. Celem algorytmu jest znalezienie elementu, rozumianego tutaj jako **wielokąt**, do którego należy wybrany punkt. Dodatkowo zaimplementowano program realizujący **graficzną reprezentację** kroków omawianego algorytmu, tak aby algorytm mógł służyć jako **narzędzie dydaktyczne** objaśniające działanie algorytmu. Przeprowadzono również **analizę wydajnościową**, aby uwidocznić **efektywność** działania zaimplementowanego algorytmu.

**2. Dane techniczne**

***Specyfikacja komputera na którym testowano algorytm: system Ubuntu 24.04.01 Linux 5.15 x64, procesor AMD Ryzen 7 5825U with Radeon 2GHz 8 rdzeni, 16GB pamięci RAM.*** Ćwiczenie zostało napisane w języku ***Python 3.9.2*0 w *Jupyter Notebook*** w środowisku programistycznym ***Visual Studio Code*.** Aby wykonać ćwiczenie posłużono się bibliotekami: ***numpy, matplotlib, json***. Do wykonania wizualizacji stworzono specjalne klasy oraz funkcje, które później zostaną szczegółowo opisane. Do procesu wizualizacja posłużono się również narzędziem graficznym przygotowanym przez Koło Naukowe BIT.

**Wstęp teoretyczny dlaczego nie ta metoda n^2 + z filmiku rzeczy ( co jest na prezentacji?)**

1. **Struktury przygotowane dla algorytmu**
2. **Opis działania algorytmu**
3. **Wizualizacja**
4. **Testy działania**
5. **Testy wydajnościowe**
6. **Podsumowanie i wnioski**
7. **Bibliografia**

**# opisać przypadek brzegowy z prostokątem**

**# opisać założenia zbioru odcinków**

**# opis mapy trapezowej i inne struktury**

**Dokumentacja**

1. **Uruchamianie**

Aby poprawnie uruchomić program należy uruchomić plik .ipynb w środowisku bitalg, na przykład przy użyciu aplikacji Jupyter Notebook. Do poprawnego działania program niezbędne są stosunkowo aktualne wersje programów i bibliotek, zostało to opisane w pliku requirements.txt. Aby program działał prawidłowo należy uruchomić wszystkie komórki kodu dostępne w pliku, zalecane jest kompilowanie pliku w ustalonej kolejności.

1. Przykład uruchomienia
2. Spis używanych bibliotek
3. Klasy co robią i metody
4. Funkcje i algorytmy