

Dokumentacja

December 6, 2019

1 Folder *utils*

Zawiera struktury oraz algorytm wykorzystywane w programie

1.1 `data_structures.py`

Plik zawierający implementacje struktur wykorzystanych w projekcie

- **Klasa Vertex**
 - Opis
 - * Przechowuje dane wierzchołków
 - * Zawiera metodę samosortującą incydentne krawędzie po kącie
 - Zmienne
 - * x, y - współrzędne
 - * *hedgelist* - lista półkrawędzi incydentnych do wierzchołka
- **Klasa Hedge**
 - Opis
 - * Przechowuje dane ścian
 - Zmienne
 - * origin - wierzchołek końcowy
 - * twin - bliźniacza półkrawędź skierowana przeciwnie
 - * face - ściana leżąca po lewej
 - * nexthedge - półkrawędź następnik
 - * angle - kąt ułożenia półkrawędzi względem OX
 - * prevhedge - półkrawędź poprzednik
 - * lenght - długość półkrawędzi
- **Klasa Face**
 - Opis
 - * Przechowuje dane ścian
 - Zmienne
 - * wedge - jedną z półkrawędzi otaczających ścianę
- **Klasa DCEL**
 - Opis

- * Przechowuje dwukierunkowy łańcuch odsyłaczowy krawędzi
 - * Zawiera metodę `build_dcel` tworzącą DCEL
- Zmienne
 - * `vertices` - lista wierzchołków
 - * `hedges` - lista półkrawędzi
 - * `faces` - lista ścian
 - * `el` - lista krawędzi
 - * `vl` - lista wierzchołków
- **Klasa Event**
 - Opis
 - * Przechowuje dane potrzebne przy algorytmie zmiatania
 - * Zawiera porównywacz dla struktury stanu z algorytmu zmiatania
 - Zmienne
 - * `position` - współrzędne
 - * `event_type` - typ zdarzenia (początek, koniec lub przecięcie odcinka)
 - * `line` - odcinek do którego należy
- **Klasa Line**
 - Opis
 - * Przechowuje dane potrzebne przy algorytmie zmiatania
 - * Zawiera metody potrzebne w algorytmie zmiatania
 - Zmienne
 - * `start, end` - współrzędne
 - * `m, b` - współczynniki prostej

1.2 sweep.py

Plik zawierający algorytm zmiatania.

Importuje zbiory *sortedset* and *sortedlist* z pakietu *blis*.

Importuje klasy *Line* and *Event* z pliku *data_structures.py*.

- **Wyznaczanie przecięć krawędzi**

```
def get_intersections
```

 - Opis
 - * Wyznacza punkty przecięć krawędzi i dodaje je do podziału
 - Argumenty
 - * `List` - Lista odcinków będących krawędziami
 - * `e` - epsilon, dokładność porównywania wyznacznika do zera
 - Zwraca
 - * Listę punktów będących wierzchołkami początkowych krawędzi i przecięciami krawędzi

- * Listę krawędzi wraz z krawędziami dodanymi w miejscu przecięć

1.3 algorithm.py

Plik zawierający algorytm wyznaczania nałożenia dwóch podziałów Importuje metodę *get_intersections* z pliku *sweep.py* Importuje klasę *DCEL* z pliku *data_structures.py*

- **Wyznaczanie nałożenia dwóch podziałów**

def *overlay_of_subdivision*

– Opis

- * Wyznacza zbiór wynikowy będący nałożeniem dwóch podziałów

– Argumenty

- * S1, S2 - listy DCEL reprezentujące 1 i 2 podział

– Zwraca

- * Listę DCEL reprezentującą nałożenie dwóch podziałów

2 visualiser.ipynb

Plik zawierający wizualizację algorytmu nakładania podziałów wizualizacja napisana jest przy pomocy biblioteki *matplotlib*, a konkretnie modułu *animation*. Polega na kolejnym wywoływaniu funkcji *animate(i)* gdzie *i* jest klatką na sekundę, w której dokonuje się aktualizacja symulacji.