Aga Patro

Dokumentacja techniczna

1. Wyznaczanie najmniejszego okręgu

Program służy do wyznaczenie najmniejszego okręgu który zawiera w sobie chmure punktów na płaszczyźnie.

Do realizacji tego programu konieczne będzie zaimportowanie bibliotek **random, math, sympy oraz matplotlib.**

Składa się on z następujących funkcji:

min_circle - która przyjmuje zbiór punktów. funkcja ta odpowiada za przygotowanie punktów do wyznaczania (unique_point_set), wywołanie najważniejszej funkcji wyznaczającej okrąg (welzl_algorithm) oraz za wizualizację okręgu (draw_circle).

unique_point_set - usuwa ona zdublowane punkty w zbiorze. Przyjmuje zbiór punktów oraz zwraca krotkę (seen, keep) co oznacza punkty przerobione oraz punkty “zatrzymane” w zbiorze

welzl_algorithm - jest to najważniejsza funkcja w całym programie. Wyznacza najmniejszy okrąg, poprzez rekurencyjną realizację algorytmu Welzla. Przyjmuje listę unikalnych punktów *points* oraz pustą listę *R.* Zwraca krotkę która zawiera w sobie punkt będący środkiem okręgu oraz promień okręgu.

p_is_inside_circle - sprawdza czy punkt znajduje się w środku okręgu. Przyjmuje punkt w postaci krotki oraz okrąg. Zwraca flagę *False* jeśli punkt nie znajduje się w granicach okręgu lub *True* jeśli się w nim zawiera.

is_cocircular - sprawdza czy wszystkie punkty w R znajdują się na tym samym okręgu. By tak było, to ich odległość od środka okręgu musi być taka sama. Przyjmuje zbiór R. Zwraca flagę *True* jeśli wszystkie punkty z R znajdują się na jednym okręgu i *False* jeśli nie.

calculate_circle_center - Oblicza środek okręgu, poprzez obliczenie punktu przecięcia prostopadłych dwóch odcinków. Przyjmuje trzy punkty w postaci (x, y) oraz zwraca znaleziony środek okręgu (x, y).

check_all_points_inside - sprawdza czy wszystkie punkty znajdują się w środku okręgu. Przyjmuje okrąg oraz zbiór punktów. Zwraca flagę *False* jeśli nie i *True* jeśli tak.

draw_circle - odpowiada za wizualizację znalezionego okręgu. Przyjmuje okrąg który chcemy zwizualizować oraz zbiór wszystkich punktów**.**

Aga Patro

2. Wyznaczanie najmniejszych prostokątów

Program służy do wyznaczenia dwóch prostokątów zawierających w sobie chmure punktów na płaszczyźnie: prostokąt o najmniejszym polu oraz prostokąt o najmniejszym obwodzie.

Do realizacji tego programu konieczne będzie zaimportowanie bibliotek **random, math, scipy** oraz **matplotlib.**

Program składa się z następujących funkcji:

give_me_rectangle - jest to funkcja która odpowiada za przebieg programu. Przyjmuje zbiór punktów oraz typ prostokąta jaki chcemy otrzymać (o minimalnym obwodzie (0) lub polu (1)). Funkcja wyznacza punkty otoczki wypukłej za pomocą metody *ConvexHull* z biblioteki *sciPy,* następnie wywołuje funkcję algorytmu *rotating_caliper* oraz znajduje odpowiedni prostokąt w zależności od parametru *answer_type.*
Funkcja zwraca znaleziony prostokąt w postaci: [[kąt_rotacji, pole, obwód, szerokość, wysokość, min_x, max_x, min_y, max_y, punkty_wierzchołkowe, centrum_obrotu]

rotating_caliper_all_rectangles - główna funkcja programu. Znajduje wszystkie prostokąty okalające chmurę punktów. Przyjmuje punkty otoczki wypukłej, oraz zwraca listę znalezionych prostokątów postaci [[kąt_rotacji, pole, obwód, szerokość, wysokość, min_x, max_x, min_y, max_y, punkty_wierzchołkowe, centrum_obrotu].

find_minimum_area_rectangle - funkcja znajduje prostokąt o najmniejszym polu. Przyjmuje listę wszystkich prostokątów. Zwraca znaleziony prostokąt.

find_minimum_perimeter_rectangle - funkcja znajduje prostokąto o najmniejszym obwodzie. Przyjmuje listę wszystkich prostokątów. Zwraca znaleziony prostokąt.

draw_rectangle - funkcja odpowiada za wizualizacje znalezionego prostokąta wokół lokalnych punktów. Przyjmuje ona prostokąt do narysowania oraz zbiór wszystkich punktów w chmurze. Niestety ta funkcja nie działa co sprawia że nie ma możliwości wizualizacji.

Aga Patro

Dokumentacja użytkownika

1. Wyznaczanie najmniejszego okręgu

By wypróbować program wykonaj następujące kroki:

1. Otwórz plik “kod_patro_projekt.ipbyn”

2. Uruchom wszystkie komórki od początku aż do komórki o nazwie “Testowanie i wizualizacja”

3. Przejdź do komórki “Sekcje dla użytkownika, Okrąg”. Ustawione są tam szkice kodu które odpowiadają za wykonanie programu. W funkcji generate_points są ustawione wartości domyślne, jednak bez problemu można je zmienić. (UWAGA! Nie zalecane jest ustawianie n > 10 z powodu nieznanego błędu optymalizacyjnego w kodzie)

4. Teraz możesz skorzystać z jedynej funkcjonalności tego programu: generować zbiory uruchamiając komórkę z kodem i podziwiać znaleziony okrąg (gdy już w końcu się wyznaczy!)

2. Wyznaczanie najmniejszych prostokątów

By wypróbować program wykonaj następujące kroki:

1. Otwórz plik “kod_patro_projekt.ipbyn”

2. Uruchom wszystkie komórki od początku aż do komórki o nazwie “Testowanie i wizualizacja”

3. Przejdź do komórki “Sekcje dla użytkownika, Prostokąty”. Ustawione są tam szkice kodu które odpowiadają za wykonanie programu. W funkcji generate_points są ustawione wartości domyślne, jednak bez problemu można je zmienić. (UWAGA! Nie zalecane jest ustawianie n > 10000, może to wydłużyć czas obliczeń oraz przegrzać komputer)

4. Teraz możesz skorzystać z jedynej funkcjonalności tego programu: generować zbiory uruchamiając komórkę z kodem i podziwiać znalezione prostokąty przedstawione w formie:
   [[kąt_rotacji, pole, obwód, szerokość, wysokość, min_x, max_x, min_y, max_y, punkty_wierzchołkowe, centrum_obrotu]

Aga Patro

Bibliografia

1. Wyznaczanie najmniejszego okręgu

https://en.wikipedia.org/wiki/Smallest-circle_problem

https://www.geeksforgeeks.org/minimum-enclosing-circle-using-welzls-algorithm

https://www.gamedev.net/tutorials/programming/graphics/welzl-r2484/

http://www.sunshine2k.de/coding/java/Welzl/Welzl.html

https://github.com/uhuaha/smallestCircle

https://www.cs.mcgill.ca/~cs507/projects/1998/jacob/problem.html

https://www.nayuki.io/res/smallest-enclosing-circle/computational-geometry-lecture-6.pdf

2. Wyznaczanie najmniejszych prostokątów

https://en.wikipedia.org/wiki/Rotating_calipers

https://geidav.wordpress.com/tag/rotating-calipers/

https://docs.scipy.org/doc/scipy/reference/generated/scipy.spatial.ConvexHull.html

https://chadrick-kwag.net/python-implementation-of-rotating-caliper-algorithm/

https://www.statology.org/matplotlib-rectangle/

https://matplotlib.org/stable/api/_as_gen/matplotlib.patches.Rectangle.html

http://dwoll.de/rexrepos/posts/diagBounding.html

https://hypersense.subex.com/aiglossary/bounding-box/

https://openai.com/blog/chatgpt/