Władysław Nieć
Algorytmy geometryczne – ćwiczenie 3
triangulacja wielokątów monotonicznych

1. Plan/ program ćwiczenia
A. Dostosowanie aplikacji graficznej tak, aby można było zadawać proste wielokąty przy użyciu myszki, z dodatkowym zapisem i odczytem podanych wielokątów. Wielokąty powinny być zadawane w kierunku przeciwnym do ruchu wskazówek zegara.
B. Implementacja procedury sprawdzającej, czy podany wielokąt jest y-monotoniczny.
C. Implementacja algorytmu, który dla zadanego wielokąta wyszukuje wierzchołki początkowe, końcowe, łączące, dzielące i prawidłowe. Wierzchołki są odpowiednio pokolorowane zgodnie z klasyfikacją.
D. Implementacja procedury triangulacji wielokąta monotonicznego (zgodnie z algorytmem opisanym na wykładzie).
E. Przetestowanie programu na różnych zestawach danych
2.Przygotowanie Programu.
Specyfikacja komputera na którym przygotowałem program i wykonywałem obliczenia:
hp pro-book x360 11 g1 ee
4 GB pamięci RAM
procesor: Intel Celeron N3350
częstotliwość taktowania: 1100 MHz
system operacyjny : Ubuntu 20.4
Program przygotowałem jako plik Python o nazwie Niec_kod_3.py

3. Działanie programu.
Program po uruchomieniu wyświetla dwuwymiarową płaszczyznę, na której użytkownik może wprowadzić do programu wielokąt. Wielokąt ten, w celu poprawnego działania programu powinien być wielokątem prostym, a jego wierzchołki powinny być wprowadzane w kierunku przeciwnym do ruchu wskazówek zegara. Ważne, aby po wprowadzeniu wielokąta, wcisnąć przycisk następny, w celu poprawnego wprowadzenia danych, po czym zamknąć okno z płaszczyzną. Następnie program ustali, czy wprowadzony wielokąt jest monotoniczny i poinformuje o tym użytkownika w stosownym komunikacie. Jeżeli wielokąt będzie niemonotoniczny, to program skategoryzuje jego wierzchołki i wyświetli wielokąt z wierzchołkami pokolorowanymi według następującego klucza:
zielony– wierzchołek początkowy
czerwony – wierzchołek końcowy
szary – wierzchołek prawidłowy
niebieski – wierzchołek dzielący
fioletowy – wierzchołek łączący
W przeciwnym wypadku, program zapyta użytkownika, czy ten chce, aby wyświetlić wszystkie etapy triangulacji wielokąta, czy też chce wyświetlić ostateczną triangulację, czas wykonywania obliczeń oraz zapisać trójki par współrzędnych typu float wierzchołków trójkątów, na które został podzielony wejściowy wielokąt .

Przechowywanie danych triangulacji następuje u mnie w postaci trójek liczb całkowitych określających indeks wierzchołka w wypadku zwracania jedynie gotowej triangulacji oraz w postaci zbioru bez powtórzeń odcinków zapisanych jako para liczb całkowitych określających indeks końców odcinka.

Procedura przetwarzania wielokąta:

Po wprowadzeniu danych do programu, przetwarzam je do postaci, która ułatwi mi kolejne obliczenia. W funkcji getpoints zmieniam kolejność wierzchołków, tak by ich lista zaczynała się od tego, który położony jest najwyżej, w tym celu dokonuję „przesunięcia” wszystkich elementów listy w lewo o początkowy indeks najwyższego wierzchołka. Funkcja ta zwraca przetworzoną listę punktów oraz indeks punktu położonego najniżej po przetworzeniu.

Procedura sprawdzania monotoniczności wielokąta:
po takim przetworzeniu listy wierzchołków, przechodzę po niej od wierzchołka najwyższego do najniższego, sprawdzając, czy kolejne punkty położone są coraz niżej. W przypadku napotkania punktu, który jest wyżej, niż punkt przed nim zwracam Fałsz. Następnie, analogicznie, przechodzę po wierzchołkach od najniżej położonego do końca listy w poszukiwaniu wierzchołka leżącego niżej niż jego poprzednik. Jeżeli na taki natrafię, również zwracam Fałsz. Po przejściu całej listy nie napotkawszy wierzchołka zaburzającego monotoniczność wielokąta zwracam wartość Prawda.

Procedura kategoryzacji wierzchołków wielokąta niemonotonicznego:
Przechodzę po wszystkich wierzchołkach wielokąta i w zależności od położenia wierzchołka względem jego sąsiadów oraz wyznacznika określającego kąt skierowany między trzema punktami kategoryzuje je do jednego z trzech „kontenerów punktów”:
begin, end, divide, merge, default – odpowiednio wierzchołki początkowe, końcowe, dzielące, łączące i prawidłowe.
Wyniki Obliczeń
Do analizy algorytmu triangulacji wielokąta monotonicznego wybrałem odpowiednio wielokąty zawierające 10, 20 ,40 i 80 wierzchołków, asymetryczny wielokąt z większością wierzchołków po jednej stronie, oraz „choinkę” zwierającą wiele boków bliskich równoległości do osi OX.
1. Wielokąty monotoniczne:

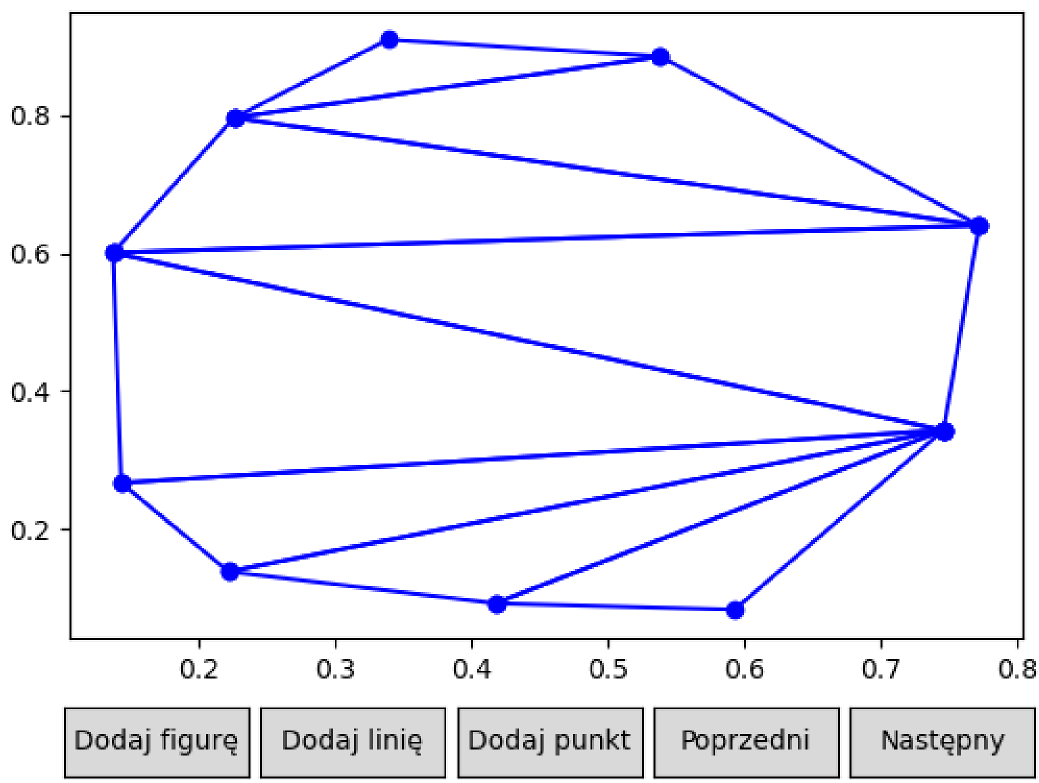

Figura 1: 10-kąt

CZAS OBLICZEŃ: 0.585 ms

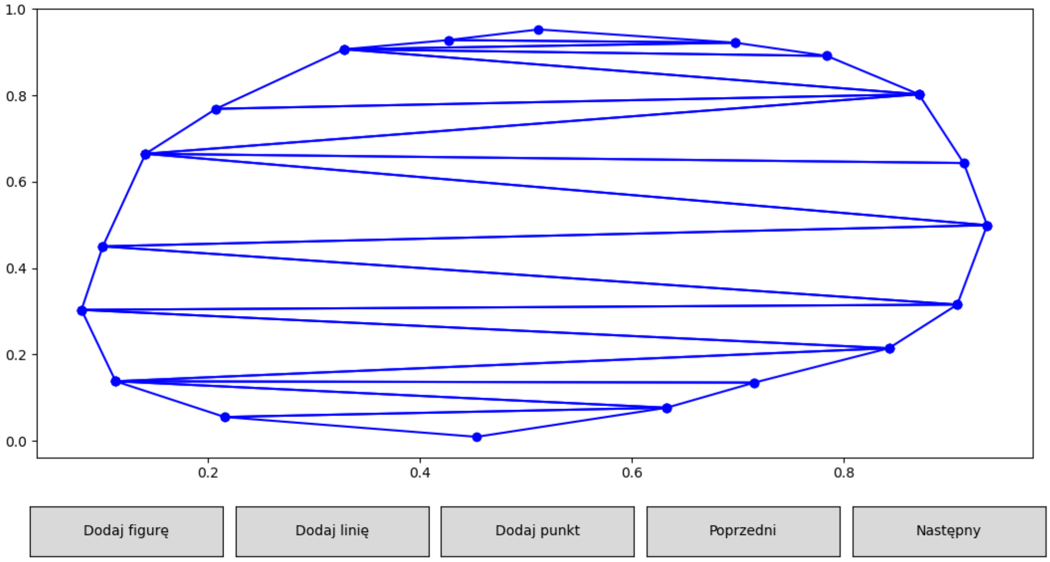

Figura 2: 20-kąt
CZAS OBLICZEŃ: 0.921 ms

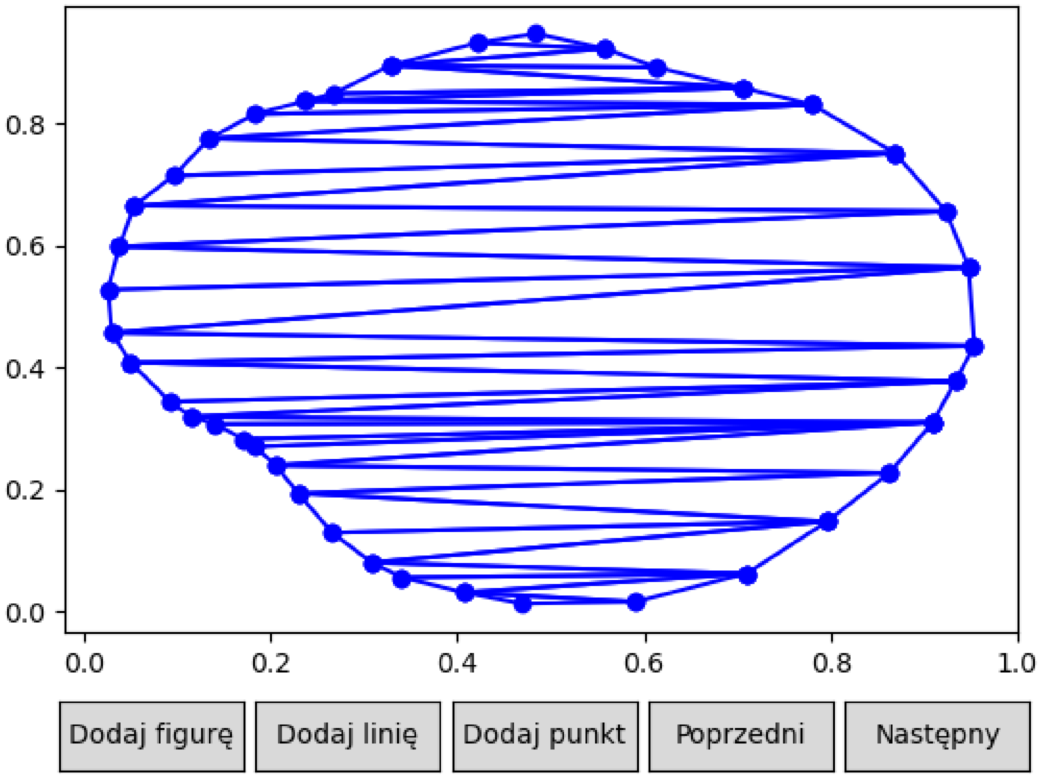

Figura 3: 40-kąt
CZAS OBLICZEŃ: 1.262 ms

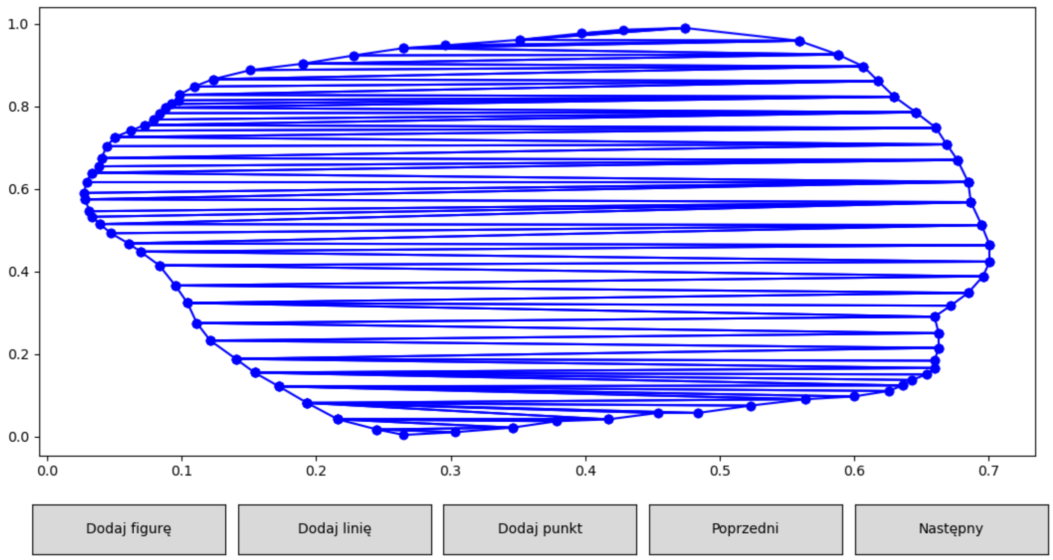

Figura 4: 80-kąt
CZAS OBLICZEŃ: 2.278 ms

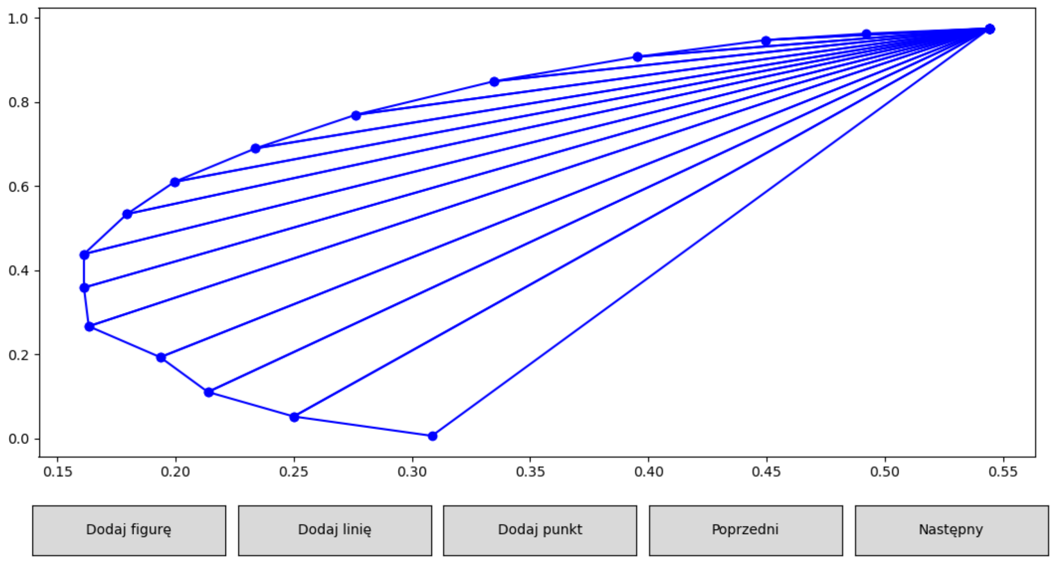

Figura 5: Wielokąt wypukły o asymetrycznej ilości wierzchołków po obu stronach

CZAS OBLICZEŃ: 0.869 ms

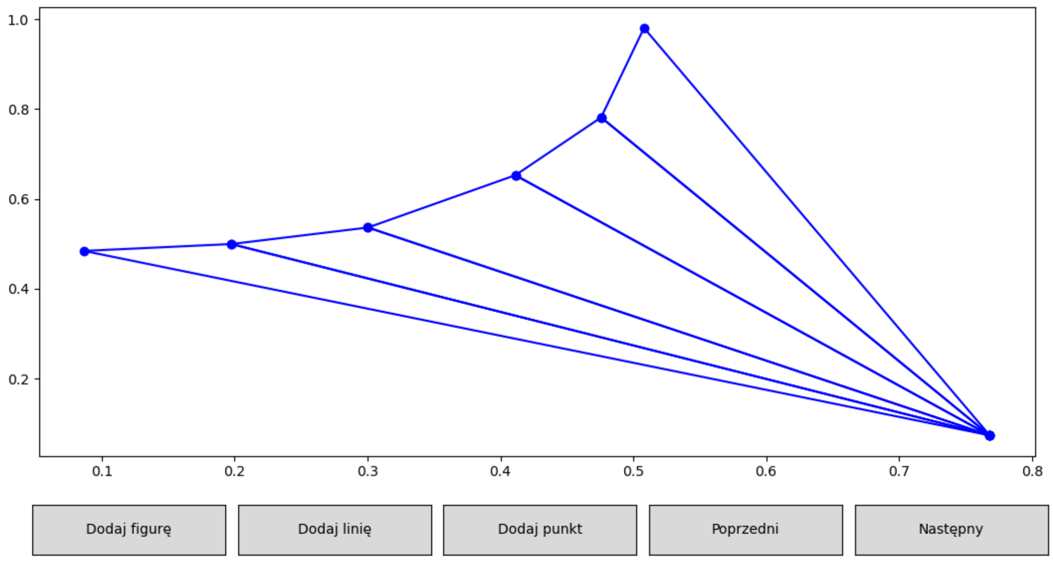

Figura 6: Wielokąt wklęsły o asymetrycznej ilości wierzchołków po obu stronach
CZAS OBLICZEŃ: 0.418 ms

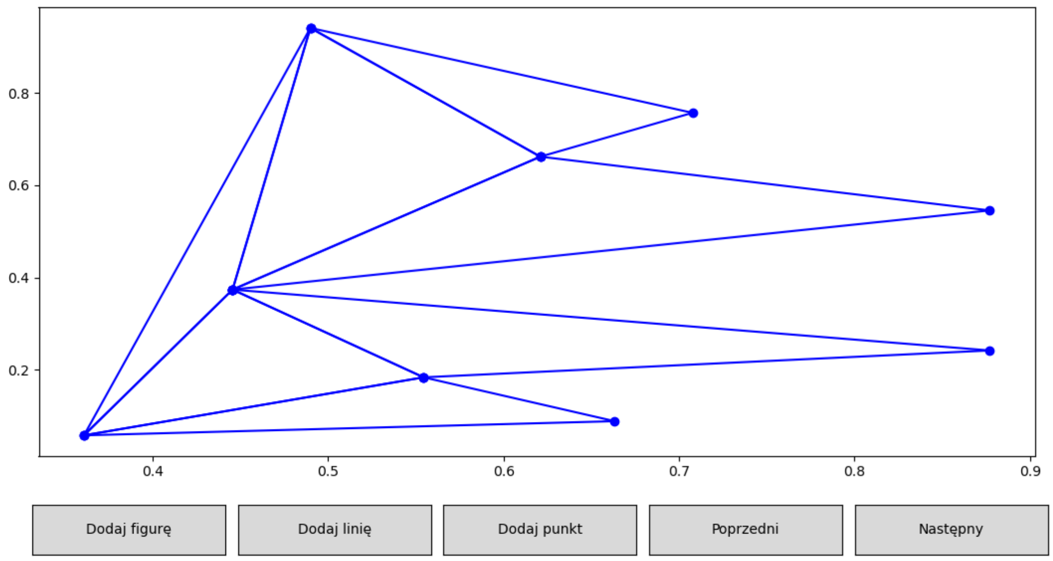

Figura 7: 9-kąt (większość wierzchołków po prawej, dolne przesłonięte)
CZAS OBLICZEŃ: 1.022 ms

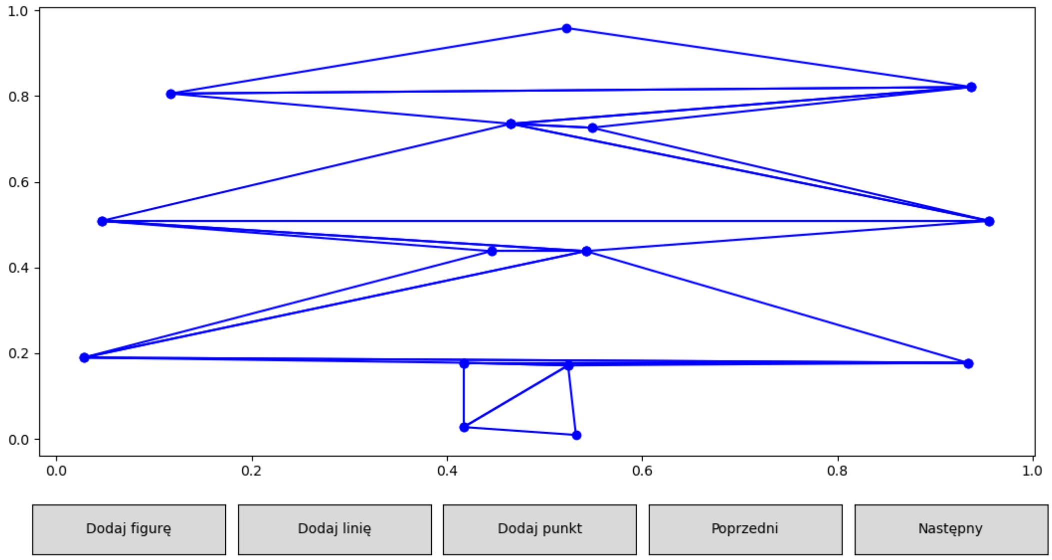

Figura 8: choinka

CZAS OBLICZEŃ: 0.962 ms

2.Etapy Triangulacji wybranego wielokąta monotonicznego

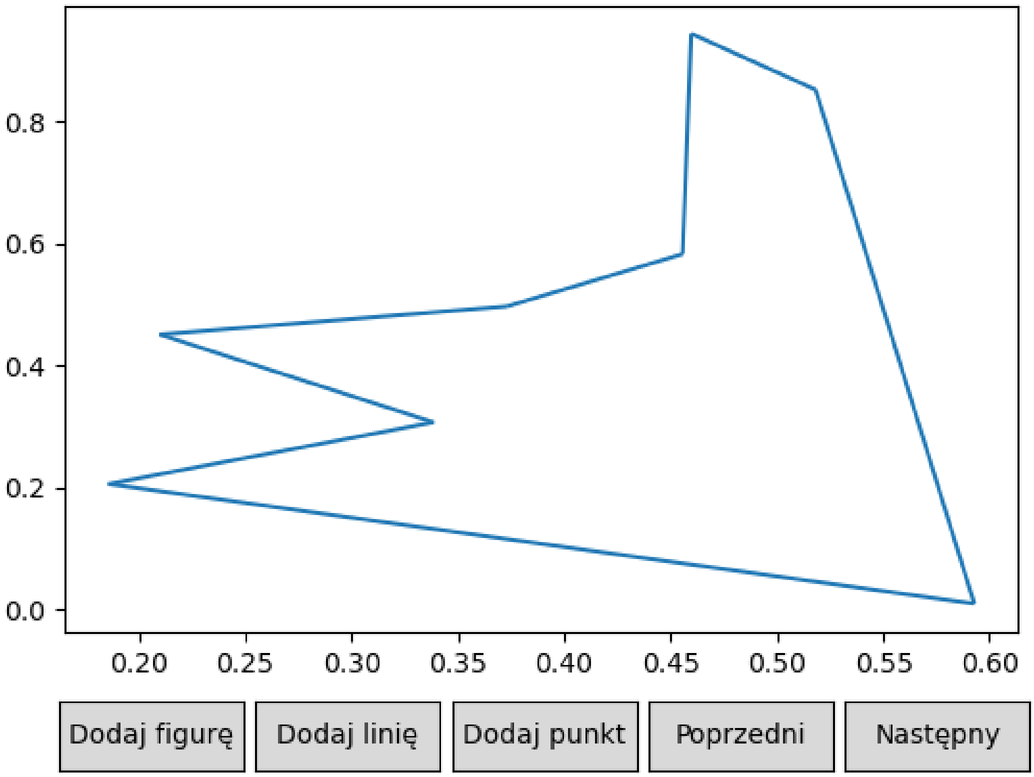

Figura 9: Zadany wielokąt

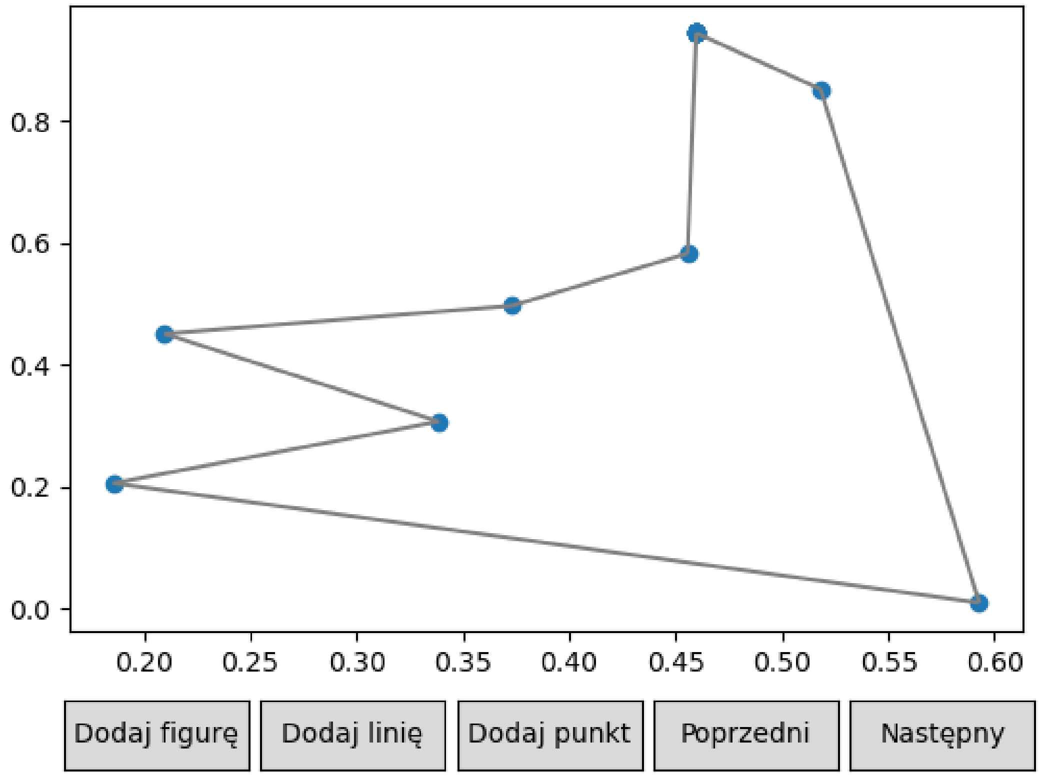

Figura 10: Etap 1

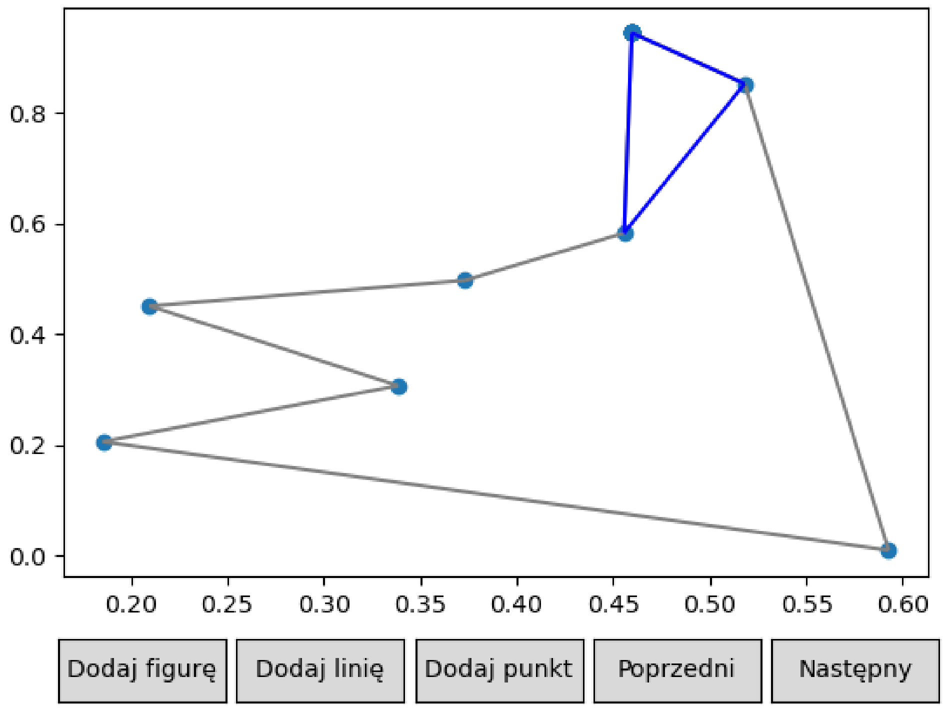

Figura 11: Etap 2

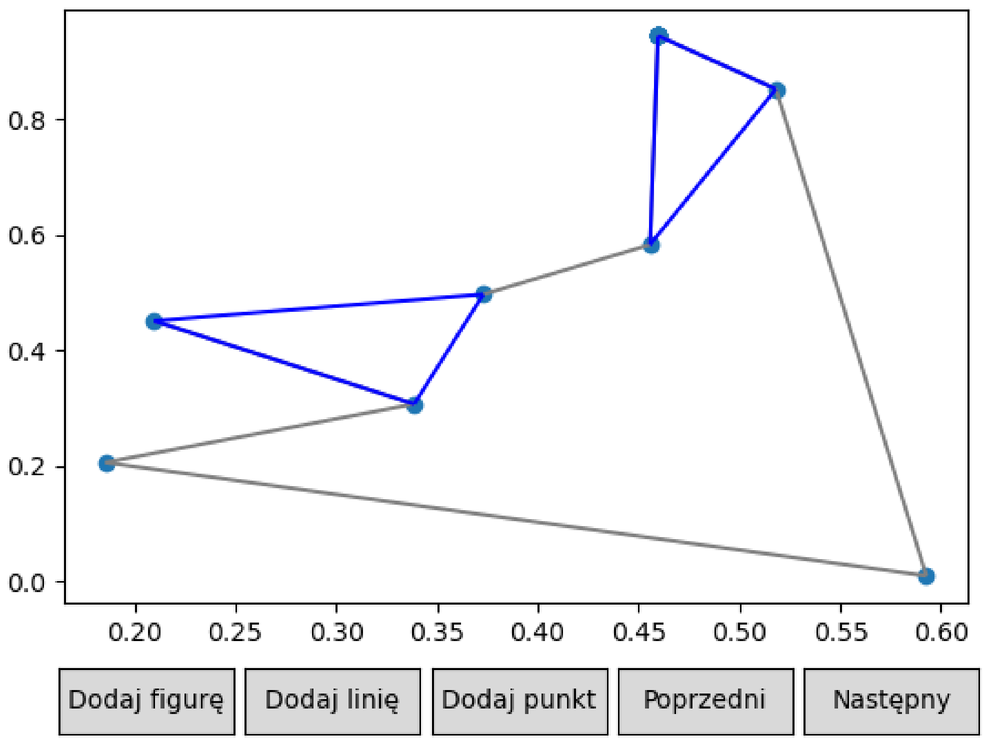

Figura 12: Etap 3

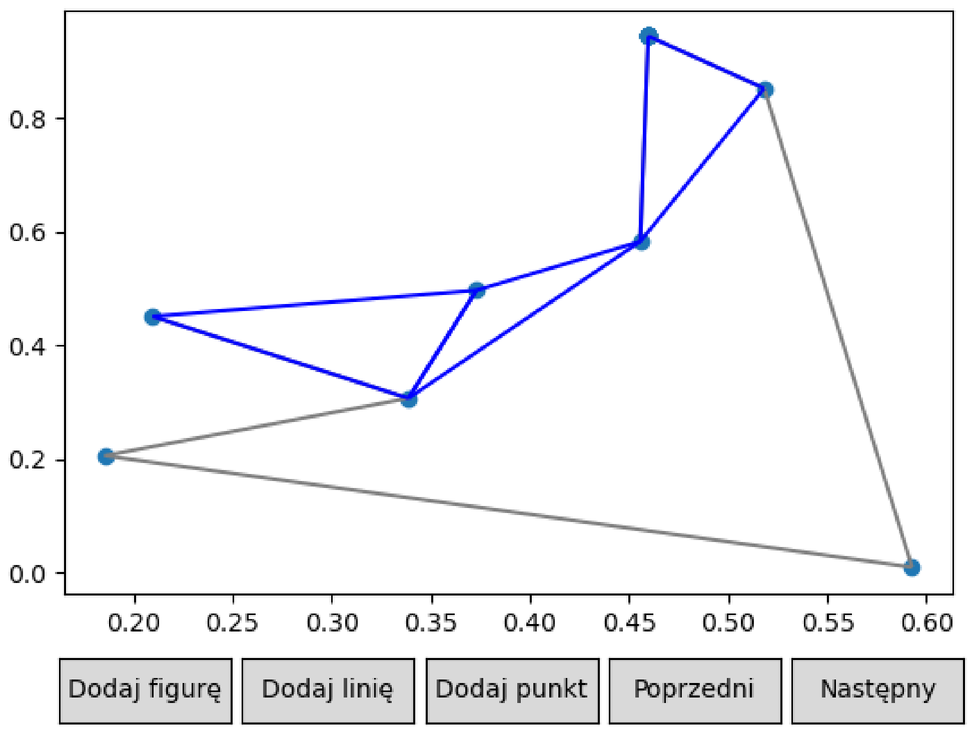

Figura 13: Etap 4

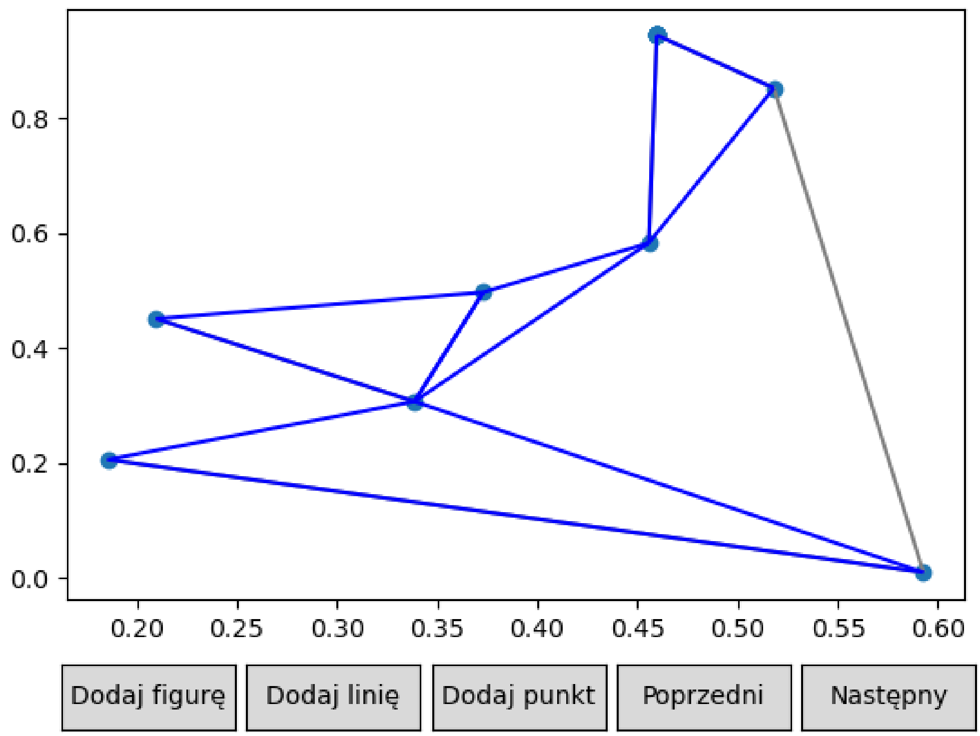

Figura 14: Etap 5

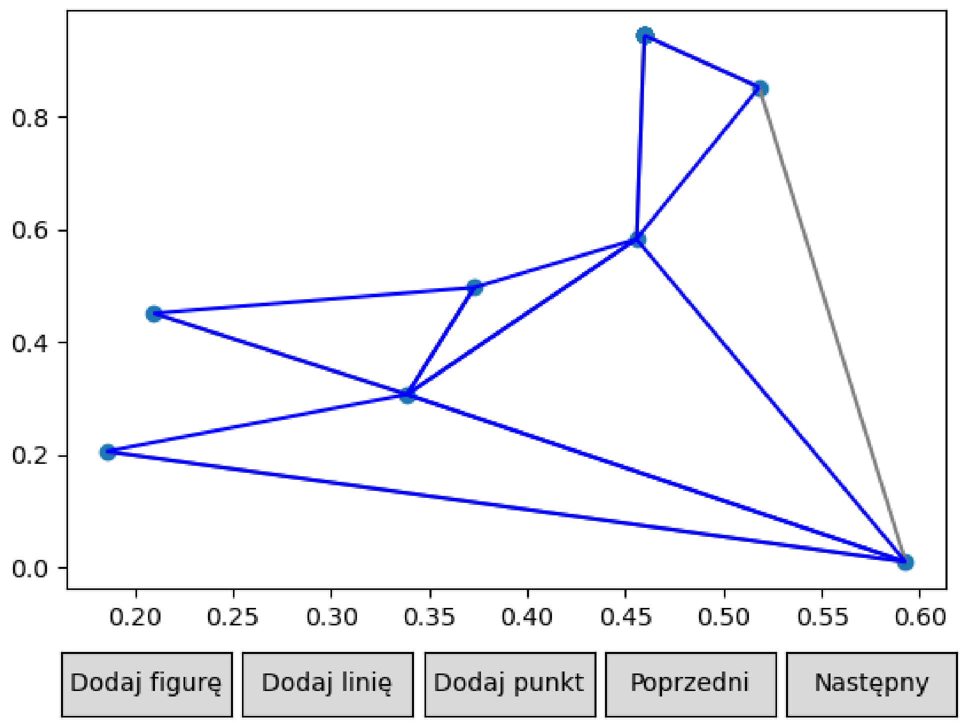

Figura 15: Etap 6

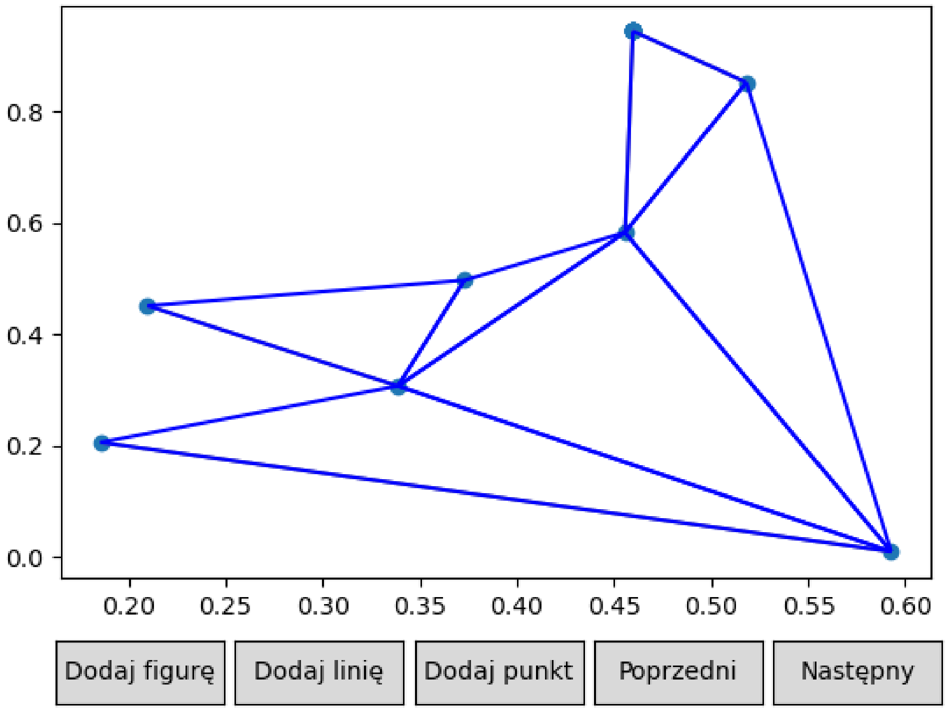

Figura 16: Etap końcowy

3. Klasyfikacja wierzchołków Wielokąta niemonotonicznego:

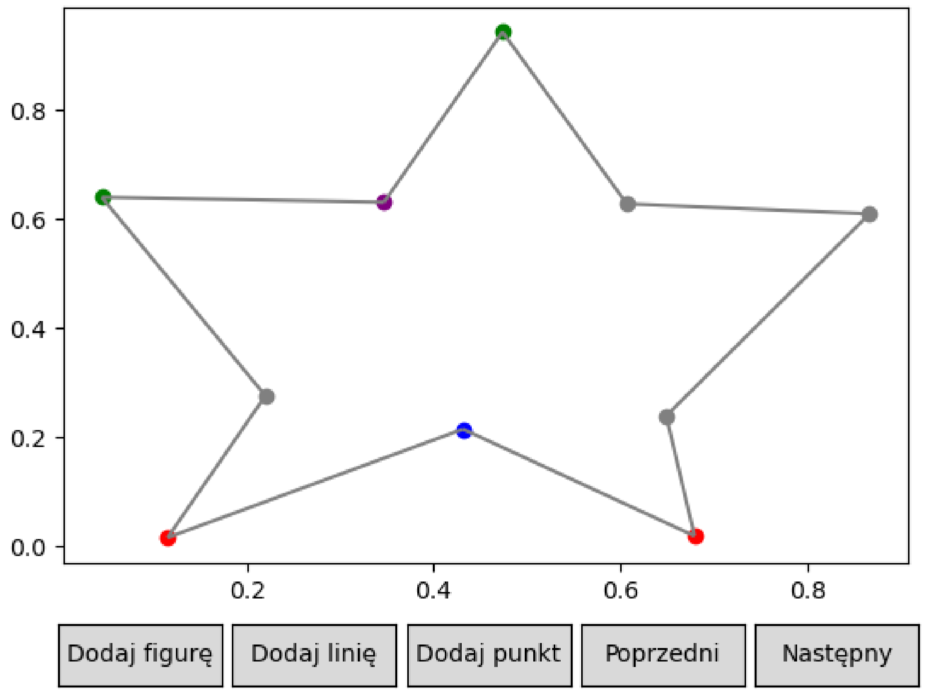

Figura 17: A) Gwiazda

CZAS OBLICZEŃ: 0.117 ms

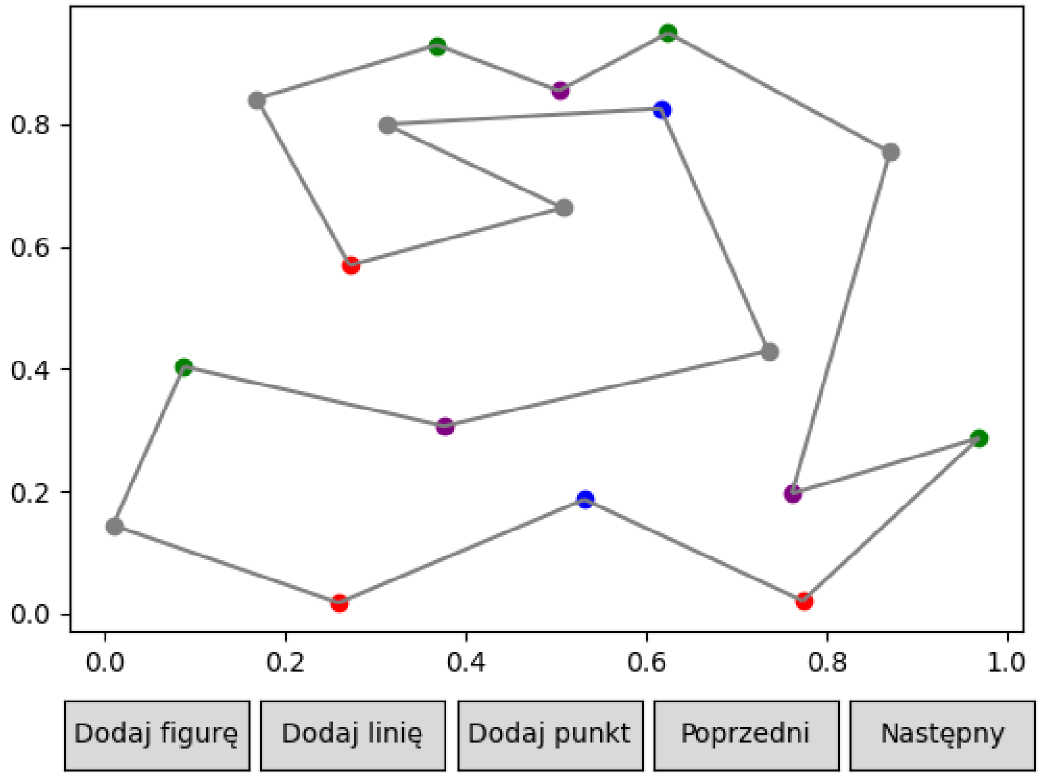

Figura 18: B) Przykładowy wielokąt 1
CZAS OBLICZEŃ: 0.169 ms

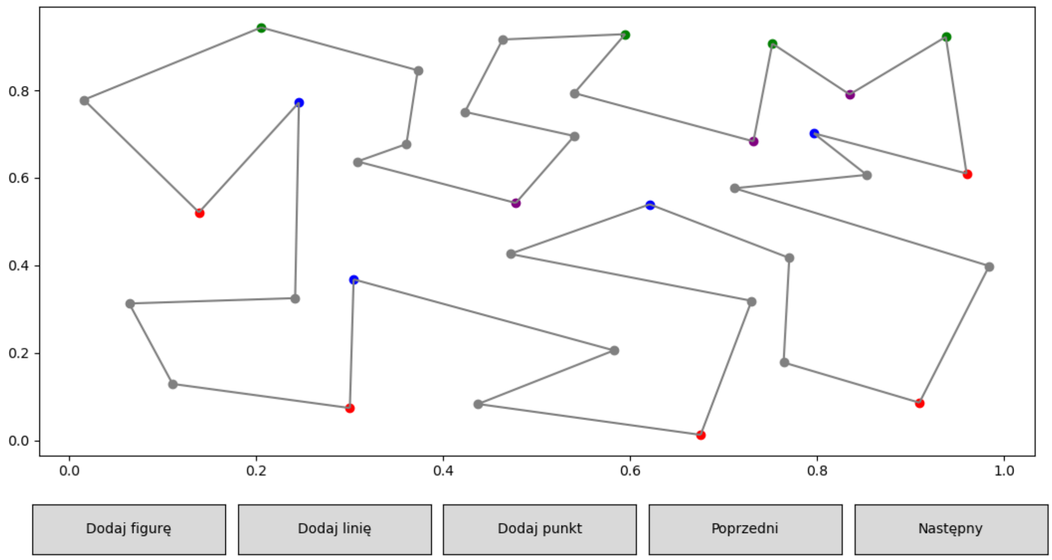

Figura 19: Przykładowy wielokąt 2
CZAS OBLICZEŃ: 0.260 ms

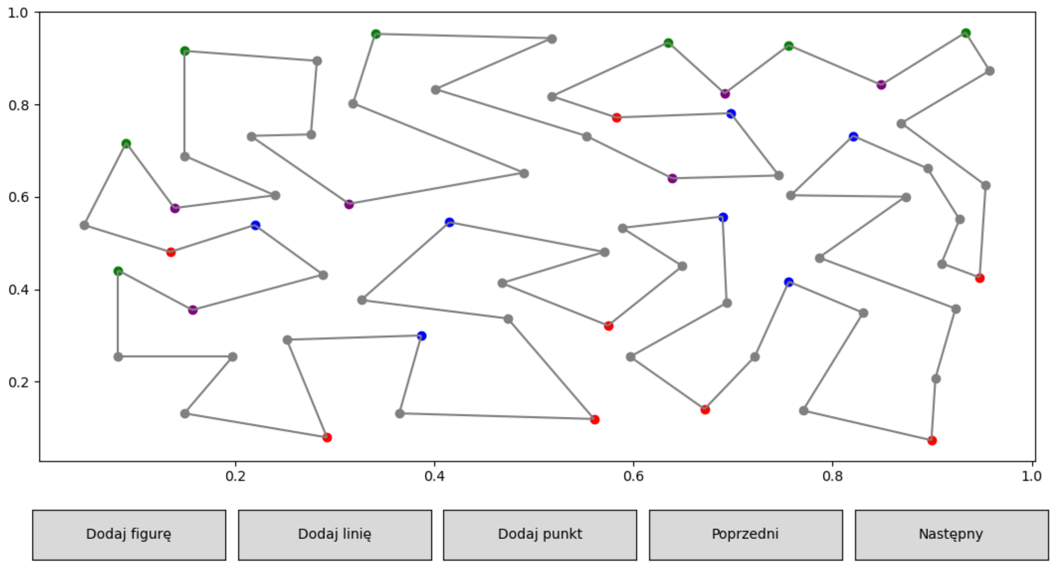

Figura 20: Przykładowy wielokąt 3
CZAS OBLICZEŃ: 0.454 ms

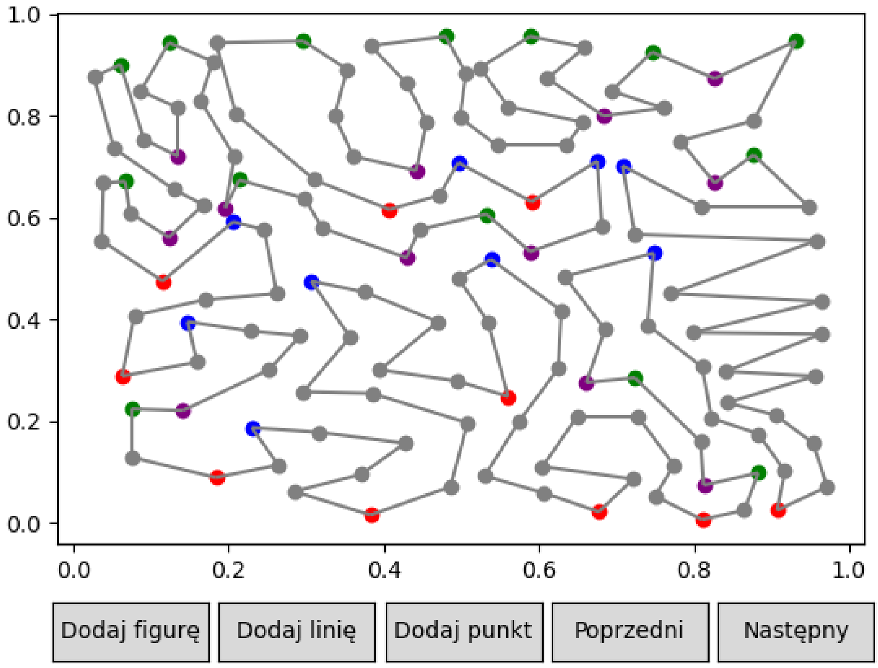

Figura 21: Przykładowy wielokąt 4
CZAS OBLICZEŃ: 0.873 ms

Tabela 1: Czasy triangulacji wielokątów monotonicznych w zależności od liczby ich wierzchołków

Liczba wierzchołków wielokąta	10	20	40	80	16 (wszystkie po jednej stronie, wypukły)	7 (wszystkie po jednej stronie, wklęsły)	9(większość po prawej, dolne przesłonięte)	15 (choinka)
Czas wykonywania obliczeń [ms]	0.585	0.921	1.262	2.278	0.869	0.418	0.1022	0.962

Tabela 2: Czasy klasyfikacji wierzchołków wielokątów niemonotonicznych w zależności od liczby ich wierzchołków

Wielokąt	A	B	C	D	E
Liczba wierzchołków wielokąta	10	18	36	70	144
Czas wykonywania obliczeń [ms]	0.117	0.169	0.260	0.454	0.873

Wnioski:
Triangulacja wielokąta monotonicznego:
Możemy zauważyć, że czas wykonywania obliczeń jest w przybliżeniu liniowo zależny od ilości wierzchołków wielokąta. Nie jest on w takim samym stopniu zależny od rozłożenia wierzchołków na „łańcuchach” wielokąta monotonicznego, ani od ilości wierzchołków o podobnej współrzędnej y.
Klasyfikacja wierzchołków wielokąta niemonotonicznego :
Możemy zauważyć, że czas wykonywania obliczeń jest również w tym wypadku statystycznie liniowo zależny od liczby wierzchołków wielokąta. Zgadza się to z budową funkcji klasyfikującej wierzchołki, złożonej z jednej pętli iterującej po wierzchołkach.