Algorytmy geometryczne laboratorium 1

Adam Dyda grupa czwartek 16:15 A

1.Cel ćwiczenia

Wprowadzenie do algorytmów geometrycznych, implementacja podstawowych algorytmów i generowania zbiorów testowych. Wizualizacja wyników działania algorytmów, opracowanie ich, i wyciągnięcie wniosków.

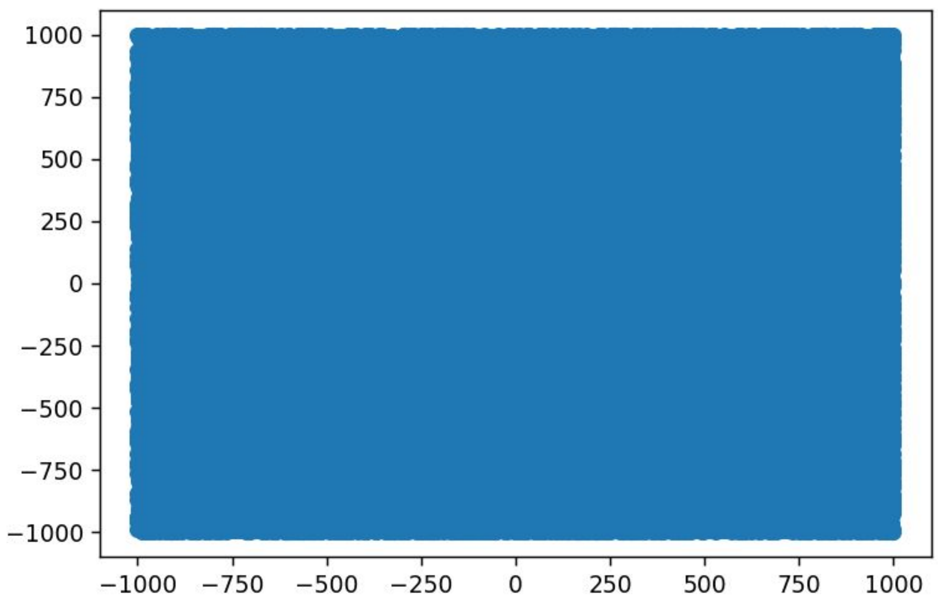
2. Wprowadzenie i przygotowanie do ćwiczenia

Do wykonania ćwiczenia wykorzystałem język python z następującymi bibliotekami, **math** - do implementacji funkcji matematycznych i losowania liczb, **numpy** - do losowania liczb i implementacji wyznacznika z funkcji linalg.det, **matplotlib** - do wizualizacji otrzymanych wyników i danych.

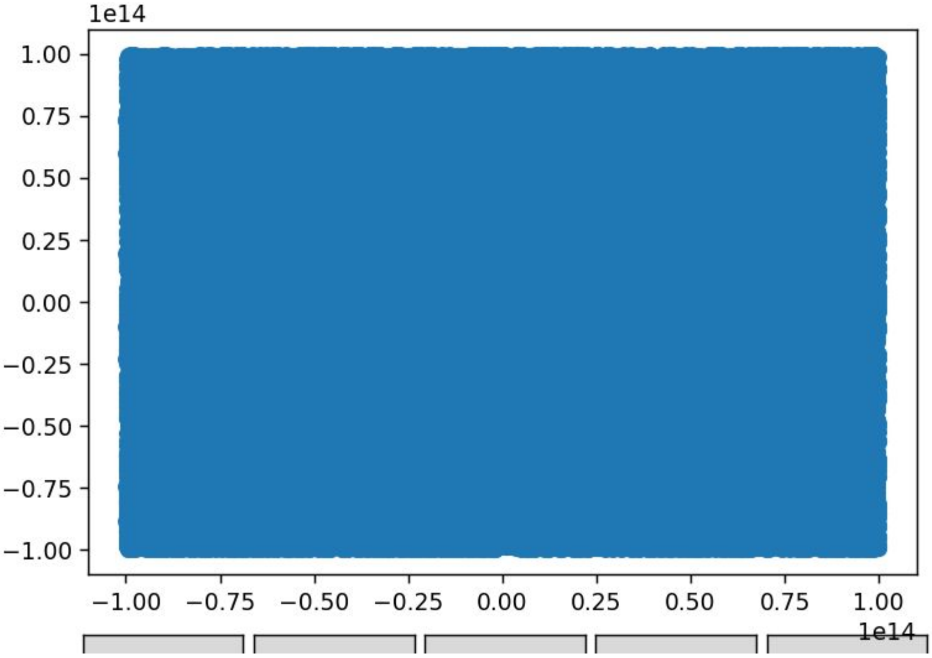
3.Wygenerowane zbiory danych na których pracowałem

zbiór A - 10^5 losowych punktów o współrzędnych z przedziału [-1000, 1000]
zbiór B - 10^5 losowych punktów o współrzędnych z przedziału [-10^{14}, 10^{14}]
zbiór C - 1000 losowych punktów leżących na okręgu o środku (0,0) i promieniu R=100
zbiór D - 1000 losowych punktów o współrzędnych z przedziału [-1000, 1000] leżących na prostej wyznaczonej przez wektor (a, b), a = [-1.0, 0.0], b = [1.0, 0.1]

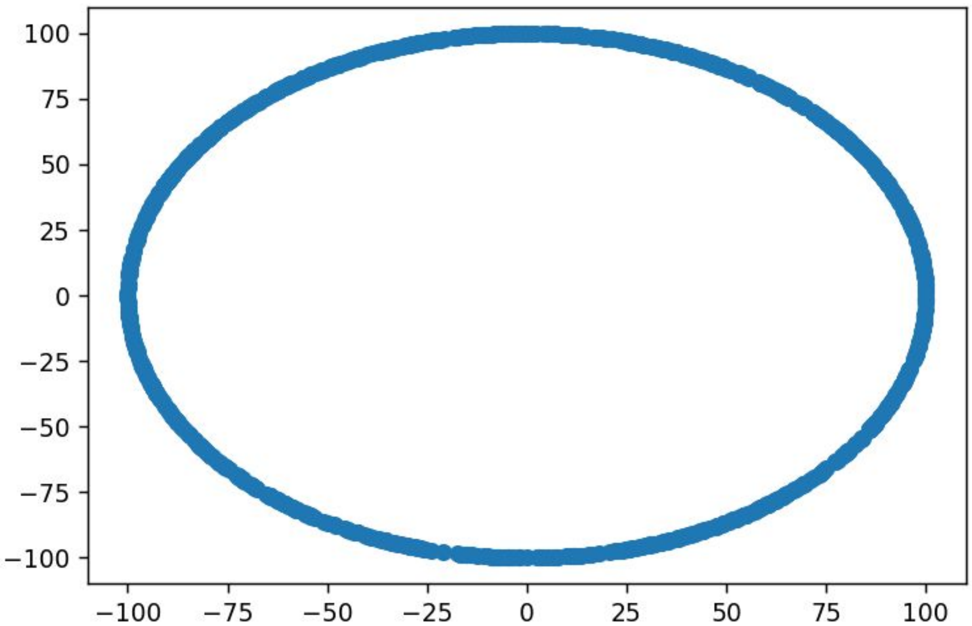
rys 1.1 zbiór A



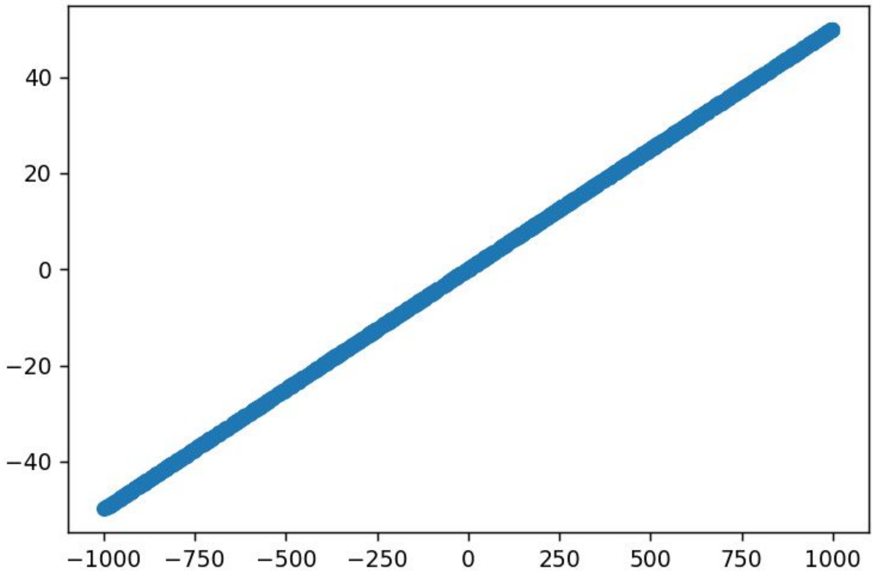
rys 1.2 zbiór B



rys 1.3 zbior C



rys 1.4 zbiór D



4. Klasyfikacja punktów dla różnych zbiorów względem wektora (a,b), a = [-1.0, 0.0], b = [1.0, 0.1]

4.1 Klasyfikacja dla zbioru A - 10^5 losowych punktów o współrzędnych z przedziału [-1000, 1000] dokładność e = 10^{-14}

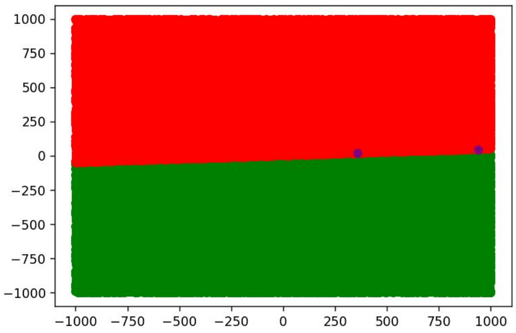
tabela 1.1

	Wyznacznik 3x3 własnej implementacji	Wyznacznik 2x2 własnej implementacji	Wyznacznik 3x3 z biblioteki numpy	Wyznacznik 3x3 z biblioteki numpy
Liczba punktów po lewej stronie prostej	49618	49619	49618	49619
Liczba punktów po prawej stronie prostej	50380	50380	50380	50380
Liczba punktów dokładnie na prostej	2	1	2	1

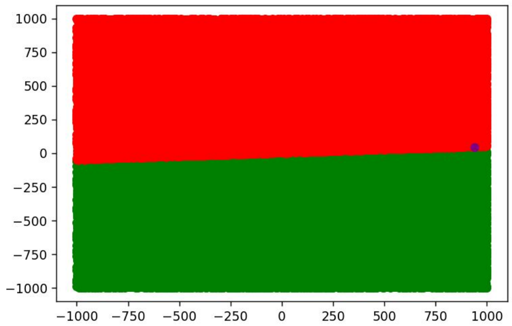
Widzimy że różnice w klasyfikacji pomiędzy wyznacznikami są niewielkie, i występują pomiędzy wyznacznikami 2x2 i 3x3.

Podział przy użyciu wyznacznika 3x3 (rys 2.1). Podział przy użyciu wyznacznika 2x2 (rys 2.2)

rys 2.1



rys 2.2



4.2 Klasyfikacja dla zbioru B - 10^5 losowych punktów o współrzędnych z przedziału [-10^{14}, 10^{14}], dokładność e = 10^{-14}

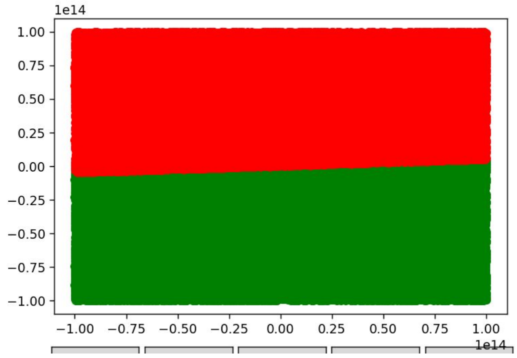
tabela 1.2

	Wyznacznik 3x3 własnej implementacji	Wyznacznik 2x2 własnej implementacji	Wyznacznik 3x3 z biblioteki numpy	Wyznacznik 3x3 z biblioteki numpy
Liczba punktów po lewej stronie prostej	49999	49995	49999	49996
Liczba punktów po prawej stronie prostej	50001	49998	50001	50000
Liczba punktów dokładnie na prostej	0	7	0	4

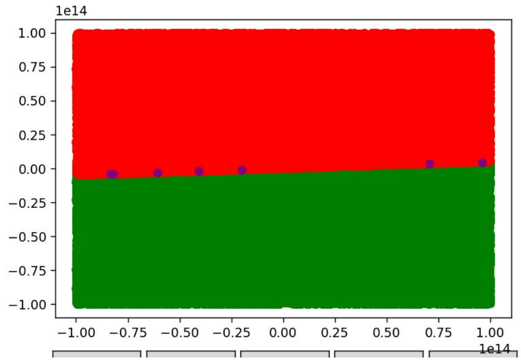
W tym wypadku, różnice pomiędzy wynikami wzrosły, prawdopodobnie jest to spowodowane, dużym zakresem w którym generowane są liczby jak i niedokładnościami związanymi z działaniami na liczbach zmiennoprzecinkowych.

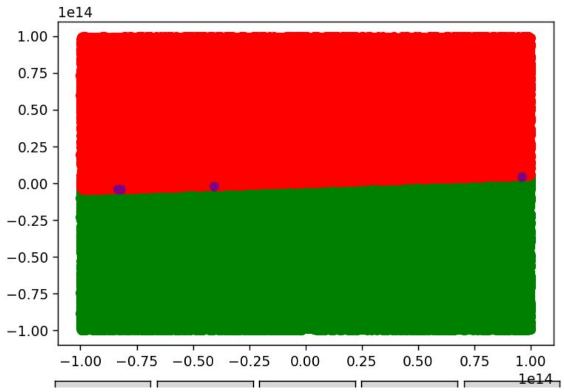
Podział przy użyciu wyznacznika 3x3 własnej implementacji (rys 2.3). Podział przy użyciu wyznacznika 2x2 własnej implementacji (rys 2.4). Podział przy użyciu wyznacznika 3x3 z biblioteki numpy (rys 2.5)

rys 2.3



rys 2.4





rys 2.5

4.3 Klasyfikacja dla zbioru C - 1000 losowych punktów leżących na okręgu o środku (0,0) i promieniu R=100, dokładność e = 10^{-14}

tabela 1.3

	Wyznacznik 3x3 własnej implementacji	Wyznacznik 2x2 własnej implementacji	Wyznacznik 3x3 z biblioteki numpy	Wyznacznik 3x3 z biblioteki numpy
Liczba punktów po lewej stronie prostej	497	497	497	497
Liczba punktów po prawej stronie prostej	503	503	503	503
Liczba punktów dokładnie na prostej	0	0	0	0

Nie ma różnic w klasyfikacji pomiędzy wyznacznikami.

Podział dla zbioru C (rys 2.6), taki sam dla wszystkich metod

rys 2.6

4.4 Klasyfikacja dla zbioru D - 1000 losowych punktów o współrzędnych z przedziału [-1000, 1000] leżących na prostej wyznaczonej przez wektor (a, b), a = [-1.0, 0.0], b = [1.0, 0.1], dokładność e = 10^{-14}

tabela 1.4

	Wyznacznik 3x3 własnej implementacji	Wyznacznik 2x2 własnej implementacji	Wyznacznik 3x3 z biblioteki numpy	Wyznacznik 3x3 z biblioteki numpy
Liczba punktów po lewej stronie prostej	0	122	98	162
Liczba punktów po prawej stronie prostej	0	131	17	143
Liczba punktów dokładnie na prostej	1000	747	885	695

W tym wypadku mamy już bardzo duże rozbieżności w działaniu algorytmów, wynikają one z różnych dokładności danych wyznaczników, warto zauważyć że przy zmniejszeniu wartości epsilon liczba punktów klasyfikowanych na prostej będzie rosnąć.

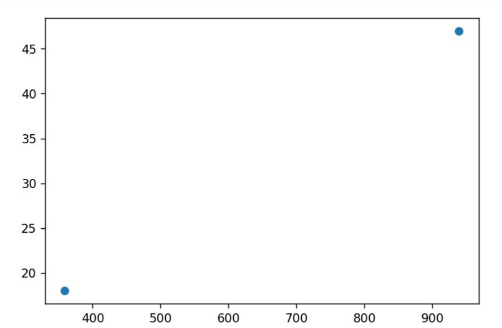
5. Porównanie algorytmu klasyfikacji dla różnych wyznaczników

Poniżej przedstawiam porównanie wyników dla różnych wyznaczników, przedstawię te różnice dla zbiorów A,B i C i tylko dla wybranych wyznaczników ponieważ tutaj różnice w klasyfikacji są małe. Natomiast zbiorem D zajmę się w następnym punkcie

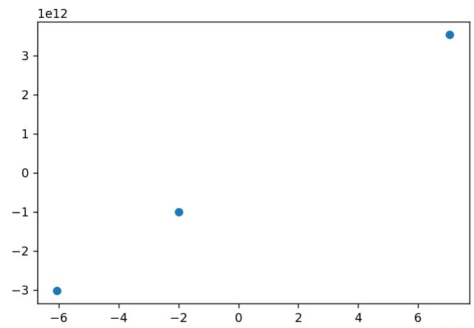
Różnica pomiędzy podziałem punktów ze zbioru A przy użyciu wyznacznika samodzielnej implementacji 3x3 a wyznacznika 3x3 z biblioteki numpy (rys 3.1)

Różnica pomiędzy podziałem punktów ze zbioru B przy użyciu wyznacznika samodzielnej implementacji 3x3 a wyznacznika 3x3 z biblioteki numpy (rys 3.2)

rys 3.1



rys 3.2



Jak widać różnice w klasyfikacji w zbiorze A i B przy przykładowych wyznacznikach są bardzo małe (2 i 3 punkty) natomiast przy zbiorze C te różnica wynosi 0 więc nie umieszczałem wykresu.

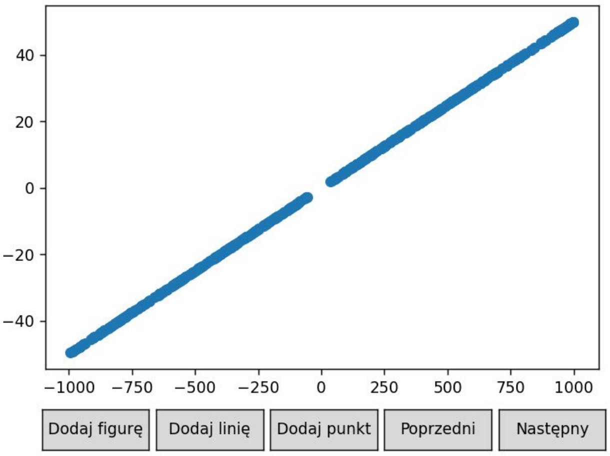
6. Porównanie w działaniu wyznaczników własnej implementacji i wyznaczników z biblioteki numpy oraz porównanie różnych tolerancji dla zera.

Poniżej przedstawiam różnice w klasyfikacji punktów ze zbioru D dla wyznacznika 2x2 implementowanego samodzielnie oraz dla wyznacznika 2x2 z biblioteki numpy. Warto zwrócić uwagę na zmianę w różnicach dla różnych wartości epsilon.

6.1 Epsilon e = 10^{-14}, liczba punktów różnie sklasyfikowanych: 402

Wykres przedstawiający punkty różnie klasyfikowane (rys 4.1)

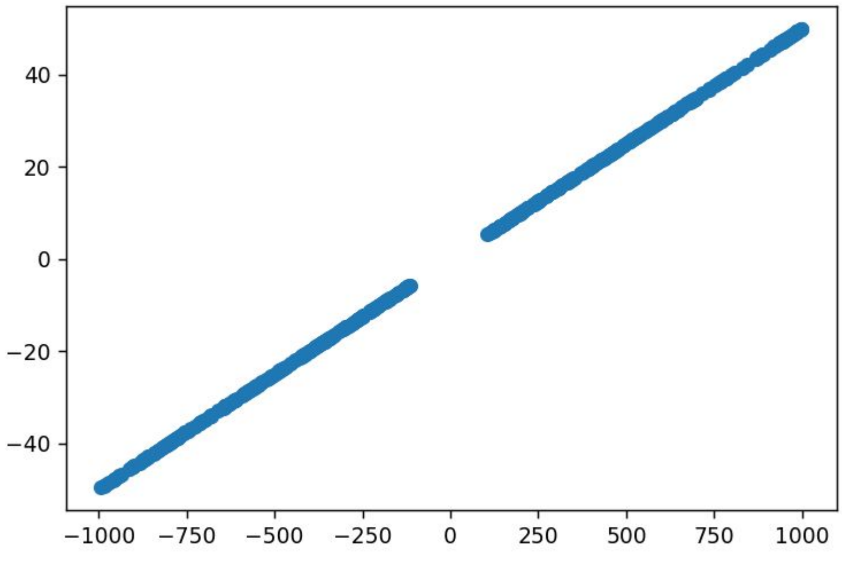
rys 4.1



6.2 Epsilon e = 10^{-13}, liczba punktów różnie sklasyfikowanych: 371

Wykres przedstawiający punkty różnie klasyfikowane (rys 4.2)

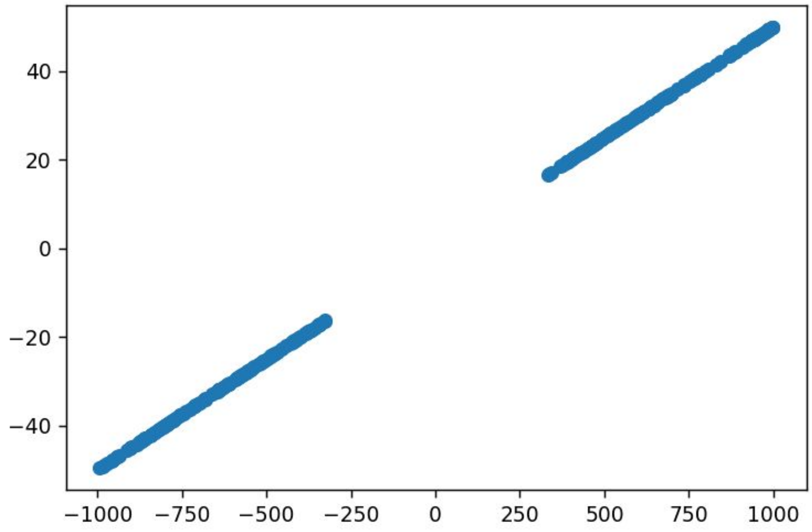
rys 4.2



6.3 Epsilon e = 10^{-12}, liczba punktów różnie sklasyfikowanych: 260

Wykres przedstawiający punkty różnie klasyfikowane (rys 4.3)

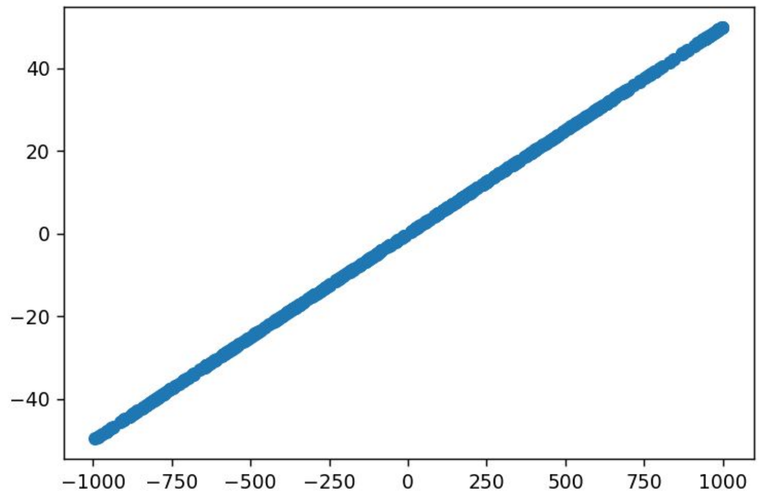
rys 4.3



6.4 Epsilon e = 0, liczba punktów różnie sklasyfikowanych: 419

Wykres przedstawiający punkty różnie klasyfikowane (rys 4.4)

rys 4.4



Widać tutaj że ze wzrostem wartości epsilon różnica w klasyfikacji punktów rośnie, wynika to z tego że wraz z maleniem epsilona rośnie obszar wokół prostej w którym punkty są klasyfikowane jako leżące na prostej.

7.Porównanie prędkości dla wyznaczników własnej implementacji a wyznaczników biblioteki numpy.

Pomiar został dokonany za pomocą biblioteki time.

Czas klasyfikacji dla zbioru A przy pomocy wyznacznika własnej implementacji : 0.1308851 s
Czas klasyfikacji dla zbioru A przy pomocy wyznacznika z biblioteki numpy : 1.9456160 s

Wynika z tego że mój wyznacznik jest szybszy, może to być spowodowany dodatkowymi operacjami jakie są wykonywane w funkcji obliczającej wyznacznik z numpy.

8.**Wnioski**

Ćwiczenie było dobrym wstępem i zapoznaniem z bibliotekami i metodami używanymi w pracy z algorytmami geometrycznymi. Można wywnioskować że dokładność danych obliczeń jest zależna od sposobu ich wykonania tak jak widzimy te różnice np. dla wyznaczników 2x2 i 3x3 oraz trzeba liczyć się z niedokładnością związaną z operacjami na liczbach zmiennoprzecinkowych. Należy także przy wykonywaniu obliczeń wziąć pod uwagę tolerancje którą przyjmujemy dla zera, przyjęcie zbyt dużej lub zbyt małej tolerancji może skutkować niedokładnością lub zakłamaniem obliczeń i wniosków