

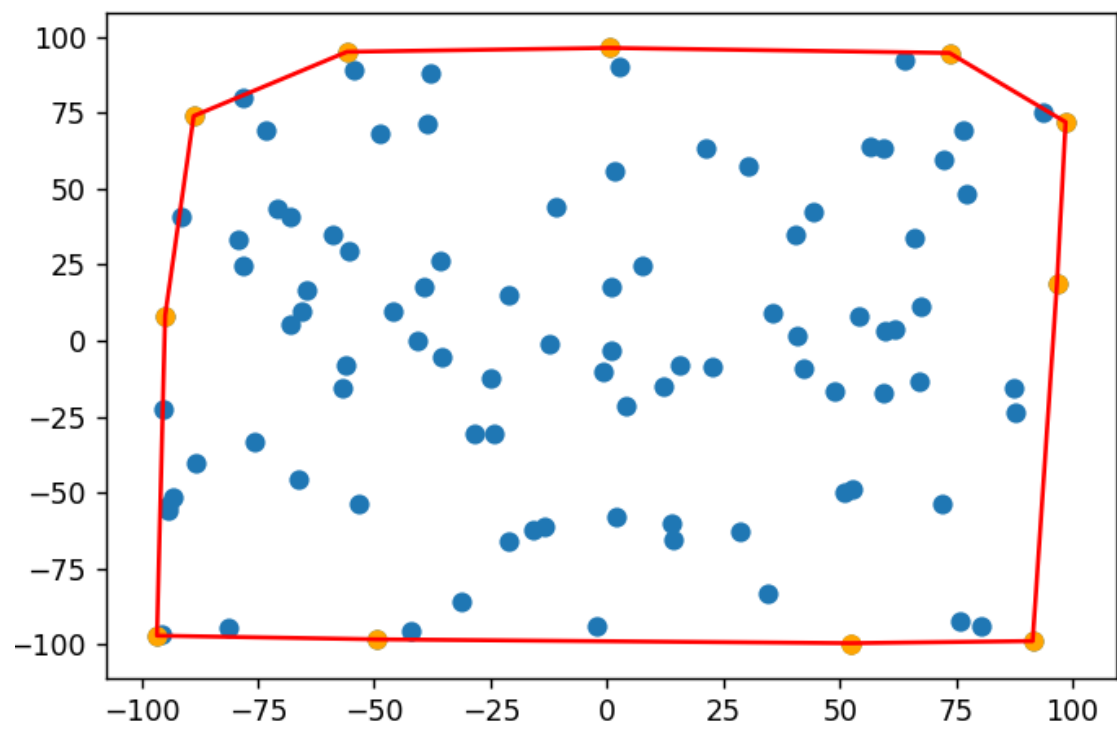
Obydwa algorytmy (Jarvisa i Grahama) działają poprawnie dla zbiorów punktów *points1, points2, points3, points4, points5, points6, points7, points8* stworzonych w sposób podany w zadaniu (Podpunkty 1 oraz 3). Zbiory *points1* oraz *points5*, składające się z losowych punktów o współrzędnych z danego prostokąta, działają poprawnie i ciężko przewidzieć ich otoczkę. Zbiory *points2* oraz *points6*, składające się z losowych punktów na danym okręgu, działają poprawnie i jak łatwo przewidzieć otoczkę to wszystkie punkty w zbiorze. Zbiory *points3* oraz *points7*, składające się z losowych punktów na danym prostokącie, działają poprawnie i składają się zawsze z 8 punktów (po 2 punkty najbardziej wysunięte na każdym boku). Zbiory *points4* oraz *points8*, składające się z losowych punktów na przekątnych kwadratu, dwóch bokach oraz czterech wierzchołków, działają poprawnie oraz jak łatwo przewidzieć otoczkę składa się zawsze z 4 wierzchołków. Każdy zbiór jest stworzony w taki sposób, by móc wyciągnąć z niego wnioski na temat działania algorytmów oraz stwierdzenie czy algorytmy działają poprawnie. Problemy, które mogły stworzyć zbiory:

- zbiory typu 2 dla algorytmu Jarvisa osiągają zawsze złożoność $O(n^2)$
- zbiory typu 3 oraz 4 porównywały niemal zawsze 3 punkty na tej samej linii, więc algorytm, który by tego nie uwzględniał zawsze byłby nieprawidłowy

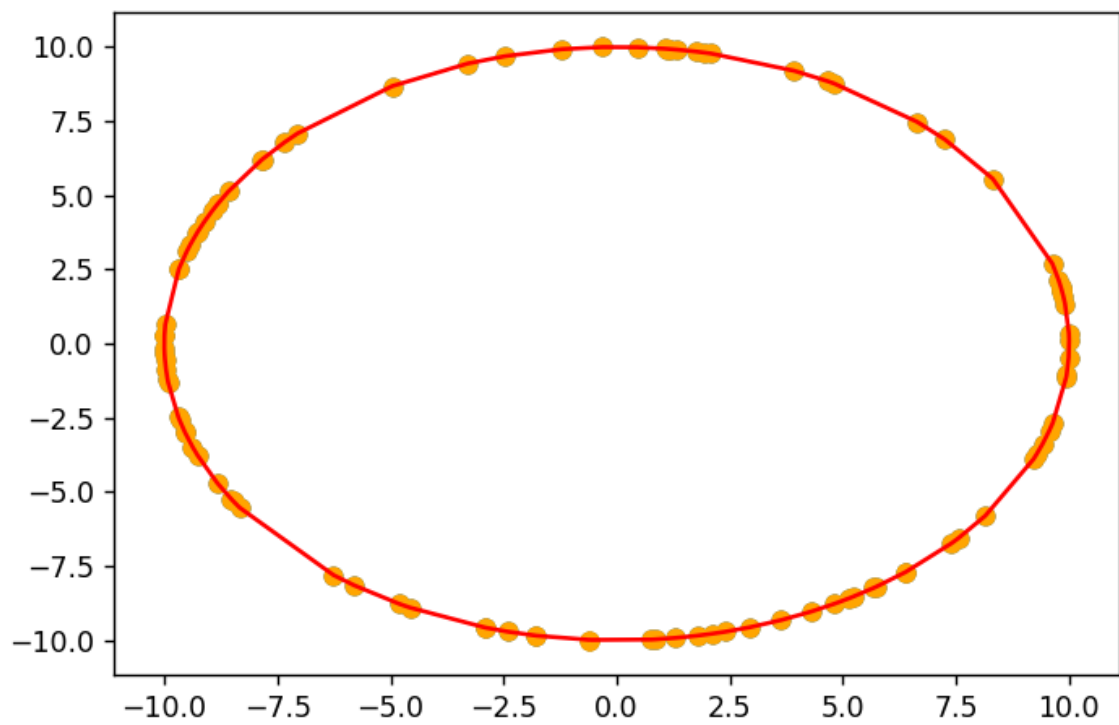
Porównanie czasów:

	Czas t (s)			
Zbiór:	Graham	Jarvis	Który szybszy?	Ile razy?
Points1	0.014002799987792969	0.004014253616333008	Jarvis	3.49
Points2	0.021941423416137695	0.11967897415161133	Graham	5.45
Points3	0.010965108871459961	0.0019943714141845703	Jarvis	5.50
Points4	0.004988193511962891	0.0	Jarvis	Bardzo dużo
Points5	0.6611931324005127	0.05186176300048828	Jarvis	12.75
Points6	1.804250955581665	15.976462602615356	Graham	8.85
Points7	59.19614887237549	0.023144960403442383	Jarvis	2557.63
Points8	2.2051022052764893	0.010970115661621094	Jarvis	201.01

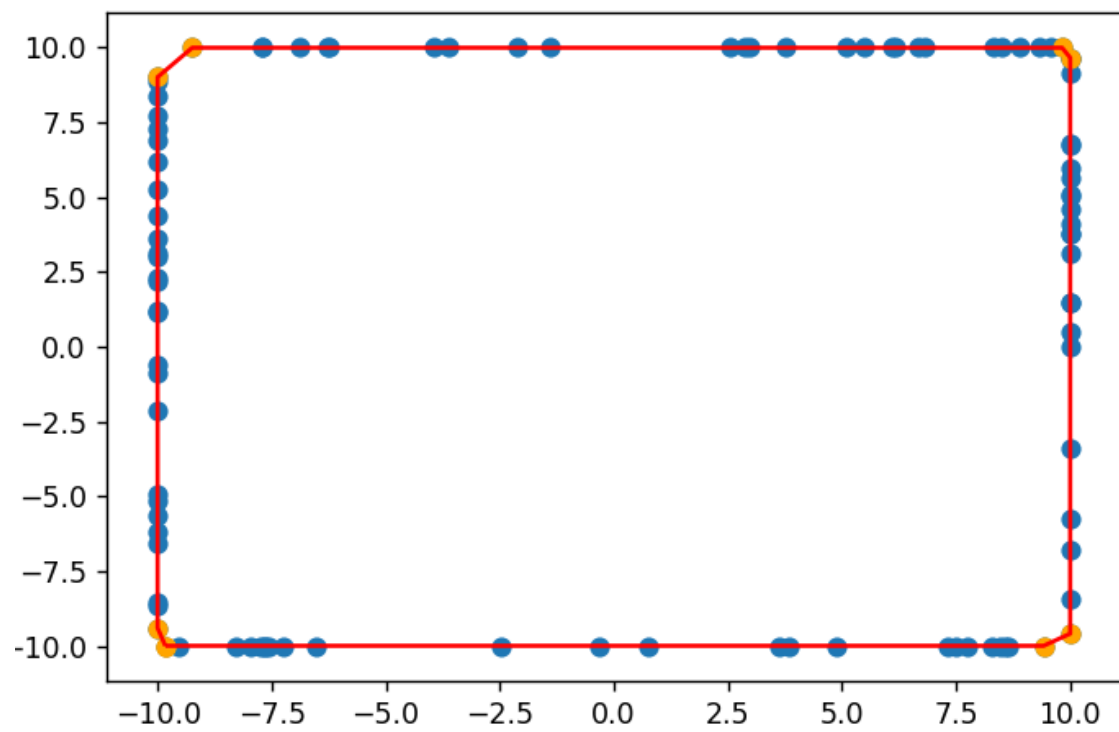
Jak widać algorytm Jarvisa jest wolniejszy tylko w przypadku, gdy osiąga złożoność $O(n^2)$. Dla dużych zbiorów osiągnął bardzo dobre wyniki w porównaniu do Grahama (nawet 2500 razy szybciej). Gdy więc wiemy, że otoczkę nie będzie duża, warto stosować algorytm Jarvisa.



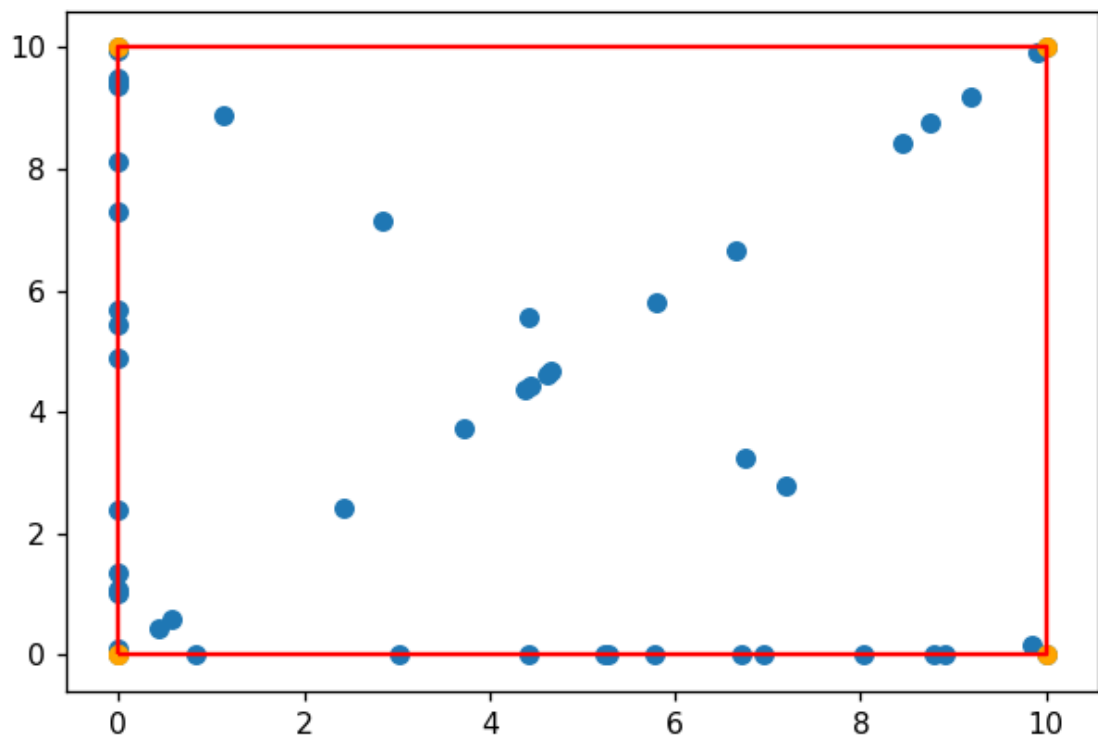
Otoczka wypukła dla zbioru *points1*



Otoczka wypukła dla zbioru *points2*



Otoczka wypukła dla zbioru *points3*



Otoczka wypukła dla zbioru *points4*