

Decyzje projektowe:

- Przekazywanie odpowiednich funkcji algorytmu poprzez klasę -> delegacja -> SRP

Wybrane stałe:

Budżet: 10k

mutations = [0.05, 0.1, 0.15, 0.2, 0.25] cross\_overs = [0.1, 0.15, 0.2, 0.25, 0.3]

- Symetria epok i populacji

Populacja 100 + epoki 100

	<b>CrossOver</b>	<b>0.1</b>	<b>0.15</b>	<b>0.2</b>	<b>0.25</b>	<b>0.3</b>
Mutacja						
0.05		971.6	1203.28	1318.12	1667.68	1783.56
0.1		1552.28	1203.72	1666.28	1783.44	1901.6
0.15		1666.76	1666.52	1783.36	1204.36	1783.64
0.2		1549.72	1784.04	1552.28	1669.28	1783.68
0.25		1667.64	1783.2	1901.0	1786.24	1784.680

Populacja 500 + epoki 20

	<b>CrossOver</b>	<b>0.1</b>	<b>0.15</b>	<b>0.2</b>	<b>0.25</b>	<b>0.3</b>
Mutacja						
0.05		1666.96	1434.68	1088.28	1666.2	1668.48
0.1		1437.28	1550.88	1785.72	1785.96	1901.44
0.15		1782.68	1552.76	1901.44	1901.72	1902.4
0.2		1784.0	1899.6	1670.0	1901.28	1786.32
0.25		1667.4	1902.2	1901.72	1902.6	1901.44

Wnioski:

- Przy za niskim prawdopodobieństwie mutacji algorytm ma problem z eksploracją <- wynikiem może być -1000
- Potrzebne jest wysokie prawdopodobieństwo krzyżowania albo mutacji, żeby algorytm był w stanie konsekwentnie osiągać wysokie wyniki
- Duża populacja z małą liczbą epok daje lepsze wyniki niż bardziej zbalansowane wartości, co ma sens, ponieważ ułatwia to algorytmowi eksplorację i wyjście z przypadku kiedy rakieta się rozbija