

# Sprawozdanie ćwiczenia nr 1 – Łukasz Szydlik

## Cel i opis eksperymentów

Celem ćwiczenia jest zaimplementowanie algorytmu ewolucyjnego z mutacją, selekcją ruletkową, krzyżowaniem oraz sukcesją generacyjną. Następnie wykorzystać go do wyznaczenia najbliższej trasy w problemie komiwojażera na zbiorze polskich miast (data/cities.csv). Należy również znaleźć ustawienie hiperparametrów algorytmu, które zwraca przyzwoite wyniki oraz zbadać wpływ wielkości populacji na proces optymalizacji.

## Instrukcja programu

### Instalacja

1. Należy pobrać repozytorium
2. Wejść w folder lab2
3. Upewnić się, że posiadamy wymagane biblioteki:

```
'pip install -r requirements.txt'
```

### Uruchomienie programu

Pamiętamy o tym aby program uruchamiać bezpośrednio z folderu lab2.

Program należy uruchomić za pomocą komendy:

```
'python main.py --cities-path="data/cities.csv"'
```

W wyniku otrzymamy długość znalezionej drogi oraz wypisaną tą drogę.

Możemy wywołać main.py z dodatkowymi parametrami:

```
--start={wybrane_miasto} - Ustawi miasto początkowe
```

```
--finish={wybrane_miasto} - Ustawi miasto końcowe.
```

```
--make-new-pops=True
```

Utworzy 100 losowych osobników dla populacji i zapisze ich w pliku „data/pops.json”. Taką ilość osobników ma program domyślnie.

Nie jest zalecane tworzenie własnoręcznie listy osobników w pliku „data/pops.json”, jednakże jest to możliwe. Należy pamiętać, że w każdej liście muszą być wszystkie liczby całkowite z zakresu 0-50. 0 ma być pierwsze a 50 ostatnie. Liczba list jest dowolna.

W pliku „data/parameters.json” można ustawić parametry algorytmu ewolucyjnego.

Domyślne najlepsze parametry: "crossing\_probability": 0.6, "mutation\_probability": 0.2,

"max\_iterations": 100

Funkcja „make\_test()” oraz plik „make\_test.py” służyły do przeprowadzania eksperymentów.

## Wyniki ćwiczenia

Na początek najważniejsze było znalezienie najlepszych parametrów prawdopodobieństw mutacji i krzyżowania. Zrobiono to na mniejszej liczbie miasto co pozwoliło na przeprowadzenie dużej ilości testów. Wybrano ten który miał względnie najlepszy średni wynik i odchylenie standardowe. Prawdopodobieństwo krzyżowania = 0.6, mutacji = 0.2

	seed=1	Ciechanów	Olsztyn	Cities=12		
Tests	Population	Crossing Probability	Mutation Probability	Iterations	AVG Value	STD
51	100	0.9	0.9	100	2329.70	80.90
51	100	0.9	0.6	100	2321.60	84.60
51	100	0.9	0.3	100	2302.55	105.82
51	100	0.9	0.2	100	2259.88	125.94
51	100	0.9	0.1	100	2267.56	130.84
51	100	0.9	0.05	100	2294.72	126.60
51	100	0.9	0.01	100	2357.97	141.64

Tests	Population	Crossing Probability	Mutation Probability	Iterations	AVG Value	STD
51	100	0.8	0.9	100	2345.77	100.01
51	100	0.8	0.6	100	2316.36	97.25
51	100	0.8	0.3	100	2298.91	113.22
51	100	0.8	0.2	100	2283.68	127.85
51	100	0.8	0.1	100	2273.64	128.74
51	100	0.8	0.05	100	2323.66	146.46
51	100	0.8	0.01	100	2443.04	124.51

	seed=1	Ciechanów	Olsztyn	Cities=12		
Tests	Population	Crossing Probability	Mutation Probability	Iterations	AVG Value	STD
51	100	0.7	0.9	100	2346.63	74.03
51	100	0.7	0.6	100	2335.63	98.11
51	100	0.7	0.3	100	2329.59	102.48
51	100	0.7	0.2	100	2289.22	121.14
51	100	0.7	0.1	100	2286.18	99.78
51	100	0.7	0.05	100	2323.28	122.40
51	100	0.7	0.01	100	2371.98	171.62

Tests	Population	Crossing Probability	Mutation Probability	Iterations	AVG Value	STD
51	100	0.6	0.9	100	2318.97	95.04
51	100	0.6	0.6	100	2353.18	101.33
51	100	0.6	0.3	100	2313.56	97.51
51	100	0.6	0.2	100	2260.34	107.16
51	100	0.6	0.1	100	2263.43	117.21
51	100	0.6	0.05	100	2269.54	116.56
51	100	0.6	0.01	100	2343.22	147.61

Następnie sprawdzono jak wielkości populacji wpływa na proces optymalizacji.

			Crossing Probability	Mutation Probability	
Ciechanów	Olsztyn	Cities=51	0.6	0.2	
Tests	Population	Iterations	AVG Value	STD	AVG Time(s)
51	10	100	16179.09	617.06	0,68
51	10	1000	14864.73	512.58	6,66
11	10	10000	13879.65	517.65	65,52
51	100	10	15100	363.04	0,81
51	100	100	14047.85	605.72	7,00
11	100	1000	12548.41	710.85	78,32
51	1000	1	15039.21	192.39	1,34
51	1000	10	14747.29	292.76	7,47
11	1000	100	13384.67	361.22	70,28

Tests – liczba wykonanych testów

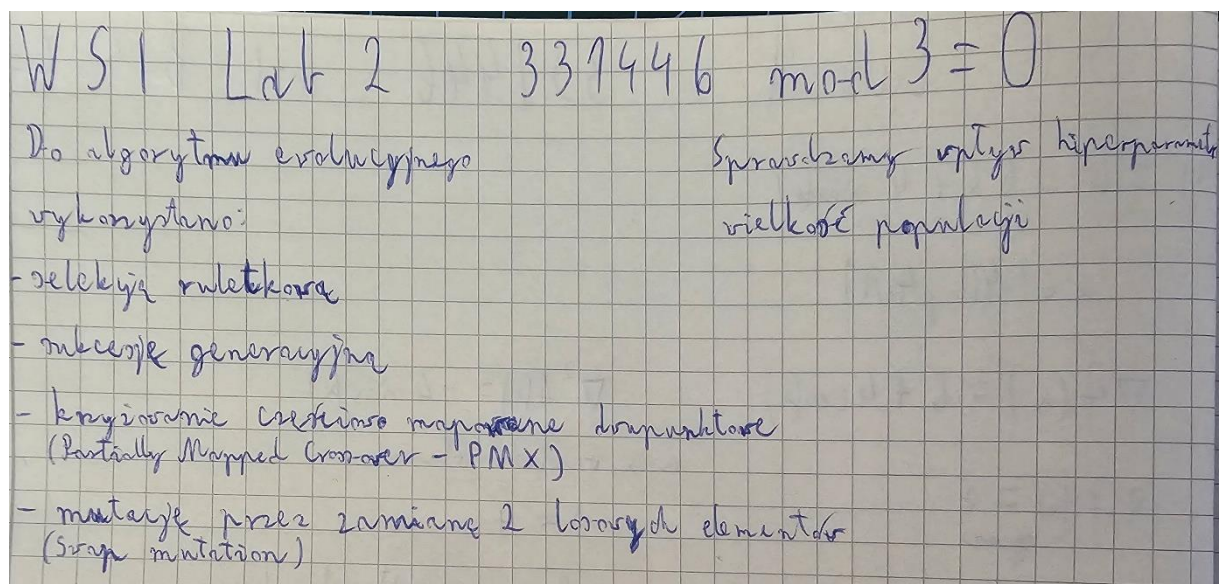
STD – odchylenie standardowe

AVG Time(s) – średni czas znalezienia najlepszego rozwiązania

Wyniki były przeprowadzane dla stałej populacji, jeśli populacja była jednakowej wielkości. Przy zmienionej wielkości populacji była na nowo losowana.

Jako punkt startowy wybrano Ciechanów, a miastem końcowym był Olsztyn.

Notes:



## Wnioski

Najistotniejszym parametrem było liczebność populacji. Im osobników było więcej tym lepszy wynik można było uzyskać. Ścisłe związane jest to z maksymalną ilością iteracji. Te dwie cechy decydują ile łącznie wykona się funkcja ewaluacyjna, czyli ocena każdego osobnika w każdej generacji. Jednakże zbyt duże wartości tych parametrów przełożą się na olbrzymi czas znajdowania najlepszego rozwiązania, który będzie spowodowany ogromną liczbą działań logicznych. Zatem parametry te trzeba dobrać w sposób odpowiedni do naszych możliwości sprzętowych i czasu, który chcemy na to przeznaczyć. Można to w łatwy sposób przewidzieć. Wystarczy początkowo ustawić algorytm dla mniejszych parametrów i uzyskać ich czas działania. Czas ten będzie rosł wprost proporcjonalnie do wzrostu liczby osobników i liczby maksymalnej iteracji.

Natomiast w kwestii pozostałych parametrów algorytmu nie zachodzi znacząca optymalizacja. Prawdopodobieństwo krzyżowania nie miało większego wpływu w znajdowaniu najlepszej drogi. Natomiast prawdopodobieństwo mutacji im było większe tym odchylenie standardowe było mniejsze.

Dużym problemem jest kwestia sposobu krzyżowania, gdyż algorytm mamy zastosować nie do prostej funkcji, a do problemu NP-zupełnego. Krzyżowanie w tym przypadku musi się odnosić do pewnej trasy. Jednakże czasem w trasie dróg wystarczy zmienić tylko dwie pozycje aby otrzymać najlepsze rozwiązanie. Nie da się tego przewidzieć.

# Wyznaczenie najlepszej trasy:

Ciechanów – Olsztyn

Obliczona odległość = 9832.945 km

Liczba miast = 51

Prawdopodobieństwo krzyżowania = 0.6, mutacji = 0.2

Liczba osobników = 1000

Liczba iteracji = 1000

```
[0, 26, 33, 48, 21, 5, 18, 47, 17, 6, 24, 3, 31, 13, 11, 19, 43, 20, 37, 39, 16, 4, 27, 49, 10, 25, 46, 23, 29, 42, 8, 9, 45, 35, 14, 1, 3, 6, 30, 15, 7, 12, 2, 34, 32, 40, 28, 44, 41, 22, 38, 50] 9832.945  
['Ciechanów', 'Ostrołęka', 'Rzeszów', 'Zamość', 'Lublin', 'Chełm', 'Krosno', 'Zakopane', 'Kraków', 'Cieszyn', 'Nowy Sącz', 'Bielsko - Biala', 'Przemysł', 'Katowice', 'Jelenia Góra', 'Legnica', 'Wałbrzych', 'Leszno', 'Słupsk', 'Szczecin', 'Koszalin', 'Bydgoszcz', 'Pila', 'Zielona Góra', 'Gorzów Wielkopolski', 'Opole', 'Wrocław', 'Łódź', 'Płock', 'Toruń', 'Elbląg', 'Gdańsk', 'Włocławek', 'Sieradz', 'Kielce', 'Biała Podlaska', 'Skierniewice', 'Poznań', 'Konin', 'Częstochowa', 'Kalisz', 'Białystok', 'Siedlce', 'Radom', 'Tarnobrzeg', 'Piotrków Trybunalski', 'Warszawa', 'Tarnów', 'Łomża', 'Suwałki', 'Olsztyn']
```

