

## Практична робота № 7

### Варіант 13

#### Дослідження мурашиних алгоритмів

**Мета:** використовуючи спеціалізовані бібліотеки та мову програмування Python навчитися дослідити метод мурашиних колоній.

#### Хід роботи:

**Завдання 7.1:** Завдання. Дослідження мурашиного алгоритму на прикладі рішення задачі комівояжера

При розробці програми було використано мережу Інтернет для пошуку можливих реалізацій необхідних методів з огляду на наданий приклад мовою Matlab.

#### Фрагмент лістингу:

```
class CityMap:
    def __init__(self, distances_matrix, cities_count):
        self.distances = distances_matrix
        self.numberOfCities = cities_count
        self.pheromones = [[np.random.rand() for j in range(cities_count)] for i
in range(cities_count)]

    def upd_pheromones(self, evaporation_rate, pheromone_delta):
        for i, row in enumerate(self.pheromones):
            for j, col in enumerate(row):
                self.pheromones[i][j] *= (1 - evaporation_rate)
                self.pheromones[i][j] += pheromone_delta[i][j]

class Ant:
    def __init__(self, city_start):
        self.startingCity = city_start
        self.currentCity = city_start
        self.distance = 0
        self.visitedCities = [city_start]

    def move(self, city_new, distance):
        self.currentCity = city_new
        self.visitedCities.append(city_new)
        self.distance += distance
```

					ДУ «Житомирська політехніка».22.121.13.000 – Лр07		
Змн.	Арк.	№ докум.	Підпис	Дата			
Розроб.		Маковська О.Ю					
Перевір.		Пулеко І. В.					
Керівник							
Н. контр.							
Зав. каф.							
					Звіт з лабораторної роботи		
					Літ.	Арк.	Аркушів
						1	10
					ФІКТ Гр. ІПЗ-19-1[2]		

```

class Colony:
    maxColonyCycles = 50
    pheromoneAddition = 0.0005
    pheromoneEvaporationRate = 0.2
    pheromoneImportance = 0.01
    distanceImportance = 9.5
    antCanVisitPreviousCities = False

    def __init__(self, ants_num):
        self.numberOfAnts = ants_num

    def find_route(self, city_map, city_num):
        min_dist = float('inf')
        route = []
        for cycle in range(self.maxColonyCycles):
            pheromones_delta = [[0.0 for i in range(city_map.numberOfCities)] for
j in range(city_map.numberOfCities)]
            for antNumber in range(self.numberOfAnts):
                ant = Ant(city_num)
                while len(ant.visitedCities) < city_map.numberOfCities:
                    next_city = self.get_next_city(ant, city_map)
                    ant.move(next_city,
city_map.distances[ant.currentCity][next_city])
                    ant_dist = ant.distance +
city_map.distances[ant.currentCity][ant.startingCity]
                    if ant_dist < min_dist:
                        min_dist = ant_dist
                        route = ant.visitedCities
                    for city in range(len(ant.visitedCities) - 1):
                        pheromones_delta[ant.visitedCities[city]][
ant.visitedCities[city + 1]] += self.pheromoneAddition /
ant_dist
                    city_map.upd_pheromones(self.pheromoneEvaporationRate,
pheromones_delta)

            return min_dist, route

    def get_probabilities(self, ant, city_map):
        result = [0 for i in range(city_map.numberOfCities)]
        total_probability = 0
        for newCity in range(city_map.numberOfCities):
            if (newCity != ant.currentCity) and (self.antCanVisitPreviousCities or
newCity not in ant.visitedCities):
                probability = pow(city_map.pheromones[ant.currentCity][newCity],
self.pheromoneImportance) * pow(
1 / city_map.distances[ant.currentCity][newCity],
self.distanceImportance)
                result[newCity] = probability
                total_probability += probability
            result = [result[i] / total_probability for i in
range(city_map.numberOfCities)]
        return result

    def get_next_city(self, ant, city_map):
        probabilities = self.get_probabilities(ant, city_map)
        random_value = np.random.rand()
        for i in range(city_map.numberOfCities):
            if probabilities[i] > random_value:
                return i
            else:
                random_value -= probabilities[i]
        return -1

```

		Маковська О.Ю.			ДУ«Житомирська політехніка».21.121.13.000 - Лр 07	Арк.
		Пулеко І.В.				2
Змн.	Арк.	№ докум.	Підпис	Дата		

```

if __name__ == '__main__':
    cityMap = CityMap(distance, len(distance[0]))
    colony = Colony(len(distance[0]))
    result = colony.find_route(cityMap, 10)
    print(f"The shortest path obtained: {result[0]} km")

    cityRoutes = "Received route: "
    for i in result[1]:
        cityRoutes += cities[i]
        if i != result[1][-1]:
            cityRoutes += "->"
    print(cityRoutes)

    fig = plt.figure(figsize=(13, 13))
    plt.xticks([i + 1 for i in range(25)])
    plt.yticks([i for i in range(25)], cities)
    plt.xlabel("City Numbers")
    plt.ylabel("City Names")
    plt.title("Ant's optimal route")
    plt.plot([i + 1 for i in range(25)], result[1], ms=12, marker='*', mfc='r',
             mec='black', mew=2, color='black', ls="--")
    plt.grid()
    plt.show()

```

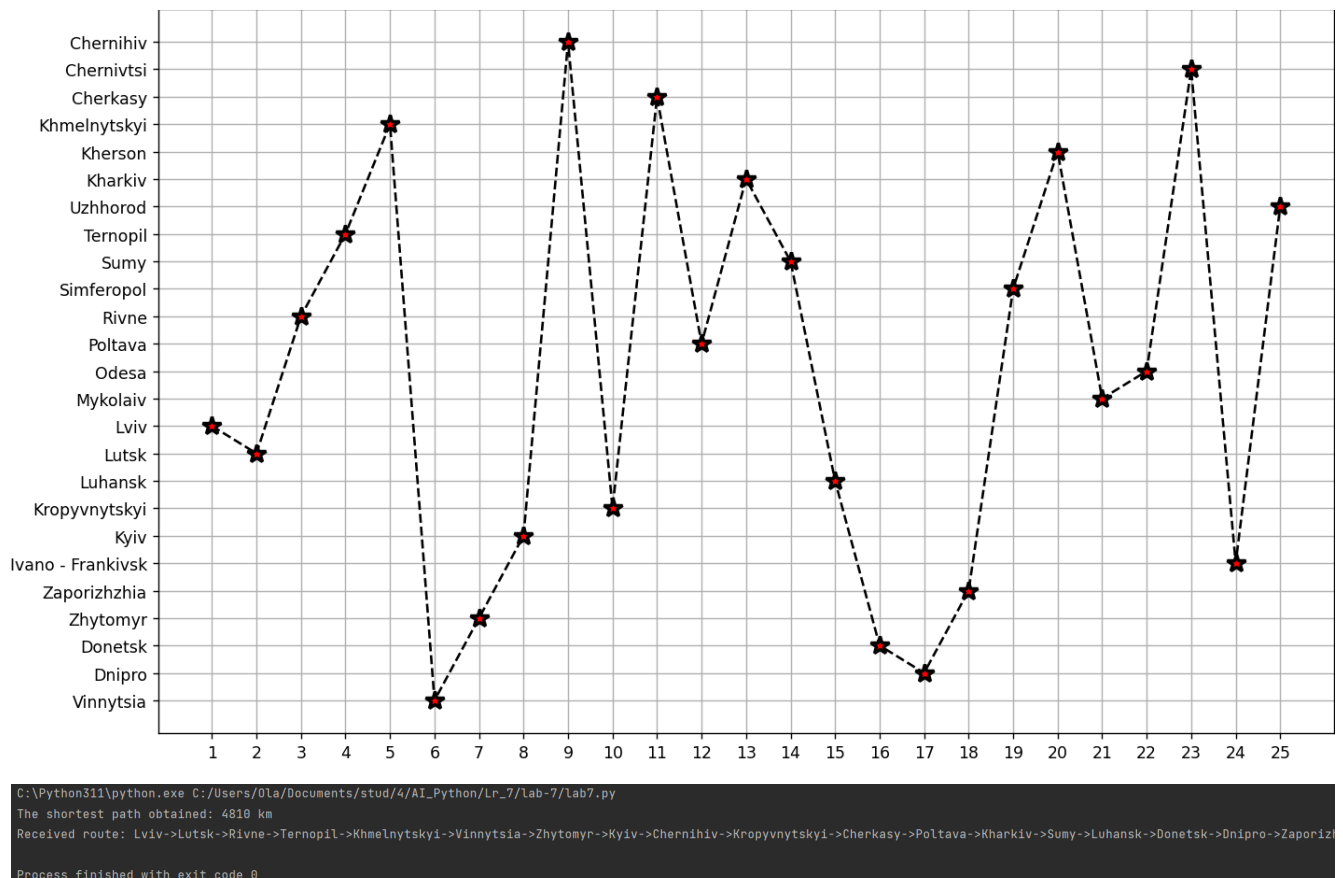


Рис.7.1. Результат виконання

		Маковська О.Ю.			ДУ«Житомирська політехніка».21.121.13.000 – Лр 07	Арк.
		Пулеко І. В.				3
Змн.	Арк.	№ докум.	Підпис	Дата		

[https://github.com/avrorilka/AI\\_Python](https://github.com/avrorilka/AI_Python)

**Висновки:** в ході виконання лабораторної роботи використовуючи спеціалізовані бібліотеки та мову програмування Python дослідили метод мурашиних колоній.

Метою цього проекту було навчитися досліджувати метод мурашиних колоній за допомогою спеціалізованих бібліотек і мови програмування Python. Завдяки дослідженням і дослідженням вдалося успішно розробити програму, яка використовує мурашиний алгоритм для вирішення проблеми комівояжера.

Мурашиний алгоритм є потужним інструментом для вирішення задач оптимізації, і його можна використовувати в багатьох різних програмах. При розробці програми в мережі Інтернет здійснювався пошук можливих реалізацій необхідних методів на основі наведеного прикладу мовою Matlab.

		Маковська О.Ю.			ДУ«Житомирська політехніка».21.121.13.000 – Лр 07	Арк.
		Пулеко І. В.				
Змн.	Арк.	№ докум.	Підпис	Дата		4