Cel ćwiczenia:

Celem ćwiczenia była analiza różnych metod rozwiązywania problemu poprzez przeszukiwanie przestrzeni rozwiązań na przykładzie problemu komiwojażera.

Ćwiczenie

Problem komiwojażera jest problemem koncentrującym się nad zagadnieniem optymalizacji. Jego głównym założeniem jest znalezienie najkrótszej drogi odwiedzającej wszystkie węzły grafu.

W celu odwzorowania tego problemu utworzyłem sieć miast, których rozmieszczenie generowane jest przypadkowo. Posiadają one takie atrybuty jak współrzędne x, y oraz nazwę (w celu pomocniczym). Następnie dodałem możliwe algorytmy ich przeszukiwania:

Metoda całkowitego przeglądu:

Jest najskuteczniejszą i najdłuższą metodą inaczej zwaną brute force. Metoda ta polega na generowaniu wszystkich możliwych permutacji zbioru (zgodnie ze wzorem n-1!), a następnie mozolnemu ich porównywaniu ze sobą, w celu wyboru najlepszego możliwego ustawienia. O ile nadaję się do prostych przykładów, to dla sieci złożonej z większej ilości węzłów (14) czas jej wykonania przekracza próg opłacalności.

1. Nie upraszcza w żaden sposób problemu, jej złożoność obliczeniowa rośnie wykładniczo.

2. Gwarantuję najlepsze rozwiązanie

3. Znajduję swoję zastosowanie gdy użycie innych metod jest niemożliwe

Metoda zachłanna:

Jest podejściem optymalnym. Ogranicza ono przestrzeń stanów poprzez zastosowanie prostej metody oceny węzłów na podstawie bieżącego kosztu. Takie rozwiązanie wybiera kolejno wszystkie najkrótsze ścieżki w grafie.

1. Jest to szybkie rozwiązanie

2. Nie gwarantuję najlepszego wyniku

3. W większości prostych przypadków jest wystarczające.

A\* :

Jest to algorytm do przeszukiwania grafów, który służy do znalezienia najkrótszej drogi z miejsca do miejsca. Opiera się na dwóch składowych. Pierwsza jest odpowiedzialna za bieżący koszt tak jak w algorytmie zachłannym. Drugą zaś jest heurystyka, która w optymistyczny sposób ocenia daną drogę mierząc jej stopień przybliżania nas do celu. Obydwie te części połączone dają bardzo dobre rezultaty. W programie podjąłem się próby zaimplementowania tego algorytmu. Niestety nie jest możliwe napisanie go do tego problemu bez stosownych modyfikacji sposobu liczenia heurystyki, bez której radzi sobie gorzej niż algorytm zachłanny.

1. Szybki i niewymagający dużej ilości czasu procesora

2. Zapewnia optymalne rozwiązanie.

3. Gwarantuje najkrótsza ścieżke