

Sprawozdanie z prezentacji algorytmu wyznaczania najkrótszej ścieżki, problem komiwojażera

Projekt numer 5 - Algorytmy i struktury danych

Autor: Krystian Petek

Problem komiwojażera - problem obliczeniowy polegający na poszukiwaniu w grafie takiego cyklu, który zawiera wszystkie wierzchołki (każdy dokładnie raz) i ma jak najmniejszy koszt.

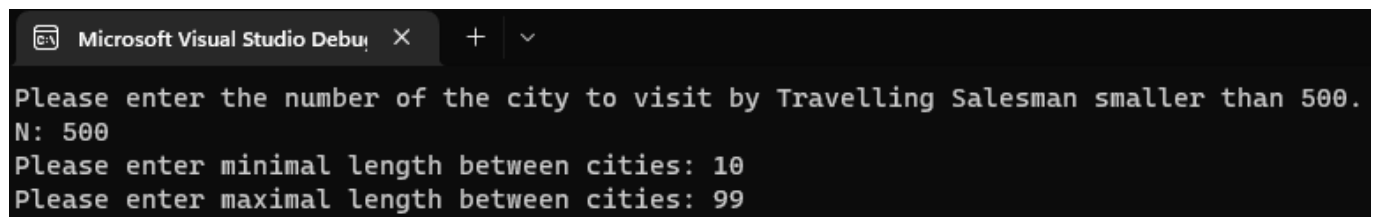
Bardziej formalnie, problem komiwojażera polega na poszukiwaniu w grafie cyklu Hamiltona o najmniejszej wadze.

Problem ma liczne zastosowania w życiu codziennym. Najlepszym przykładem jest praca kuriera, który musi wyjechać z magazynu, zawieźć przesyłki w różne miejsca i wrócić do magazynu.

1. Użytkownik jest proszony o podanie parametrów

Użytkownik podaje parametry:

- N - ilość miast do odwiedzenia (długość ścieżki): 500
- Minimalną odległość pomiędzy miastami: 10
- Maksymalną odległość pomiędzy miastami: 99



```
Microsoft Visual Studio Debug Console
Please enter the number of the city to visit by Travelling Salesman smaller than 500.
N: 500
Please enter minimal length between cities: 10
Please enter maximal length between cities: 99
```

2. Następnie program rozpoczyna działanie

1. Pierwszym krokiem podczas działania programu jest sprawdzanie czy w katalogu wyjściowym tj. `.\bin\Debug\net7.0` istnieje plik `config.json` który zawiera ostatnią konfigurację wprowadzoną przez użytkownika.
 - Jeśli istnieje to kolejnym krokiem jest sprawdzenie czy ostatnia konfiguracja różni się od wartości wprowadzonych przez użytkownika podczas aktualnego działania programu rozpoczyna się kolejny etap.
 - W przypadku braku pliku plik konfiguracyjny jest tworzony i ostatnia konfiguracja jest zapisywana do pliku.
2. Kolejnym zadaniem które jest wylosowanie odległości pomiędzy miastami w formie macierzy $[N \times N]$ w której wartości opisują dystans pomiędzy każdym z miast.
 - Wynikowa macierzy zapisywana jest w pliku `road.txt` w tym samym folderze co plik konfiguracyjny

Wycinek 30 pierwszych wartości w macierzy:

road.txt	
1	0 19 73 38 92 11 79 83 55 21 60 54 11 11 86 36 21 94 34 11 87 26 93 58 31 81 24 24 64 50
2	19 0 31 63 48 42 93 54 47 65 10 55 79 88 74 83 25 92 65 61 14 89 36 14 84 59 42 20 14 80
3	73 31 0 62 96 33 72 25 71 27 81 26 48 24 89 60 14 11 24 35 44 25 60 52 20 50 87 94 15 55
4	38 63 62 0 78 59 76 39 86 38 16 18 31 89 90 70 15 22 13 72 80 68 66 15 42 31 48 16 47 56
5	92 48 96 78 0 30 47 98 30 10 97 75 59 80 57 83 52 76 43 93 53 87 11 98 84 56 93 67 32 84
6	11 42 33 59 30 0 29 50 37 59 95 55 74 30 27 63 13 81 41 35 20 90 70 47 48 73 74 27 35 84
7	79 93 72 76 47 29 0 87 54 96 55 70 56 91 25 18 87 93 64 32 45 89 49 85 93 54 94 45 14 31
8	83 54 25 39 98 50 87 0 64 61 65 34 29 78 16 18 45 27 94 64 13 68 45 70 35 95 20 60 30 19
9	55 47 71 86 30 37 54 64 0 62 24 61 63 81 96 85 38 85 22 76 37 84 13 34 33 80 32 49 67 32
10	21 65 27 38 10 59 96 61 62 0 34 11 41 75 35 29 21 78 79 18 66 28 81 79 27 38 93 74 90 42
11	60 10 81 16 97 95 55 65 24 34 0 64 25 35 64 32 84 33 59 22 34 87 18 61 36 84 64 74 27 70
12	54 55 26 18 75 55 70 34 61 11 64 0 61 17 27 80 94 73 90 79 13 85 61 46 95 11 95 97 75 96
13	11 79 48 31 59 74 56 29 63 41 25 61 0 49 53 18 84 56 40 14 14 41 57 47 46 45 16 58 42 28
14	11 88 24 89 80 30 91 78 81 75 35 17 49 0 60 28 21 78 92 10 47 98 38 88 26 90 10 38 50 72
15	86 74 89 90 57 27 25 16 96 35 64 27 53 60 0 92 22 37 85 85 35 79 91 44 53 49 88 51 53 46
16	36 83 60 70 83 63 18 18 85 29 32 80 18 28 92 0 29 41 89 55 97 20 19 83 56 51 56 66 62 29
17	21 25 14 15 52 13 87 45 38 21 84 94 84 21 22 29 0 81 90 48 17 27 78 89 15 37 59 39 38 12
18	94 92 11 22 76 81 93 27 85 78 33 73 56 78 37 41 81 0 89 57 46 60 80 17 24 89 95 89 44 87
19	34 65 24 13 43 41 64 94 22 79 59 90 40 92 85 89 90 89 0 65 11 18 19 93 29 77 91 21 12 92
20	11 61 35 72 93 35 32 64 76 18 22 79 14 10 85 55 48 57 65 0 85 26 63 36 14 33 92 37 84 26
21	87 14 44 80 53 20 45 13 37 66 34 13 14 47 35 97 17 46 11 85 0 85 27 95 21 97 78 62 51 94
22	26 89 25 68 87 90 89 68 84 28 87 85 41 98 79 20 27 60 18 26 85 0 92 53 89 75 92 52 95 11
23	93 36 60 66 11 70 49 45 13 81 18 61 57 38 91 19 78 80 19 63 27 92 0 40 25 17 70 51 31 74
24	58 14 52 15 98 47 85 70 34 79 61 46 47 88 44 83 89 17 93 36 95 53 40 0 15 30 87 17 31 78
25	31 84 20 42 84 48 93 35 33 27 36 95 46 26 53 56 15 24 29 14 21 89 25 15 0 67 63 61 91 89
26	81 59 50 31 56 73 54 95 80 38 84 11 45 90 49 51 37 89 77 33 97 75 17 30 67 0 39 79 36 63
27	24 42 87 48 93 74 94 20 32 93 64 95 16 10 88 56 59 95 91 92 78 92 70 87 63 39 0 22 48 58
28	24 20 94 16 67 27 45 60 49 74 74 97 58 38 51 66 39 89 21 37 62 52 51 17 61 79 22 0 34 81
29	64 14 15 47 32 35 14 30 67 90 27 75 42 50 53 62 38 44 12 84 51 95 31 31 91 36 48 34 0 23
30	50 80 55 56 84 84 31 19 32 42 70 96 28 72 46 29 12 87 92 26 94 11 74 78 89 63 58 81 23 0

3. Następnie wyznaczana jest ścieżka pomiędzy miastami przez liczby kolejno od 1 do N

4. Dla wyznaczonej ścieżki między miastami losowana jest odległość z miasta do miasta

3. Dla inicjalnej drogi wyznaczonej w kroku poprzednim zostaje obliczona długość drogi od miasta 1 do N.

Wyznaczona długość dla drogi poprzez miasta od 1 kolejno do N wynosi 27269 metrów.

```
Please enter the number of the city to visit by Travelling Salesman smaller than 500.
N: 500
Please enter minimal length between cities: 10
Please enter maximal length between cities: 99
Initial road has 27269 meters.
```

4. Następnie rozpoczyna się proces poszukiwania najkrótszej długości drogi dla problemu komiwojażera

Podczas gdy znaleziono krótszą drogę przez wszystkie miasta, wynik zostaje wypisany do konsoli oraz dostajemy informacje ile kroków poczynił program aby znaleźć drogę bardziej optymalną od poprzedniej.

```
Microsoft Visual Studio Debug Console
Please enter the number of the city to visit by Travelling Salesman smaller than 500.
N: 500
Please enter minimal length between cities: 10
Please enter maximal length between cities: 99
Initial road has 27269 meters.
Found shorter road: 27142 meters in 0 attempts.
Found shorter road: 27099 meters in 1 attempts.
Found shorter road: 27082 meters in 1 attempts.
Found shorter road: 27042 meters in 1 attempts.
Found shorter road: 27034 meters in 1 attempts.
Found shorter road: 27016 meters in 1 attempts.
Found shorter road: 26901 meters in 1 attempts.
Found shorter road: 26880 meters in 1 attempts.
Found shorter road: 26797 meters in 1 attempts.
Found shorter road: 26766 meters in 1 attempts.
Found shorter road: 26672 meters in 1 attempts.
Found shorter road: 26604 meters in 1 attempts.
Found shorter road: 26539 meters in 1 attempts.
Found shorter road: 26461 meters in 1 attempts.
Found shorter road: 26428 meters in 1 attempts.
Found shorter road: 26409 meters in 1 attempts.
Found shorter road: 26407 meters in 2 attempts.
Found shorter road: 26350 meters in 1 attempts.
Found shorter road: 26333 meters in 1 attempts.
Found shorter road: 26246 meters in 1 attempts.
Found shorter road: 26241 meters in 1 attempts.
Found shorter road: 26196 meters in 1 attempts.
Found shorter road: 26170 meters in 1 attempts.
Found shorter road: 26127 meters in 1 attempts.
Found shorter road: 26063 meters in 1 attempts.
```

-
-
-

```
Found shorter road: 7106 meters in 145 attempts.
Found shorter road: 7103 meters in 33 attempts.
Found shorter road: 7096 meters in 19 attempts.
Found shorter road: 7094 meters in 28 attempts.
Found shorter road: 7092 meters in 103 attempts.
Found shorter road: 7084 meters in 79 attempts.
Found shorter road: 7082 meters in 178 attempts.
Found shorter road: 7071 meters in 11 attempts.
Found shorter road: 7069 meters in 3 attempts.
Found shorter road: 7056 meters in 47 attempts.
Found shorter road: 7040 meters in 304 attempts.
Found shorter road: 7031 meters in 436 attempts.
Found shorter road: 7025 meters in 185 attempts.
Found shorter road: 7024 meters in 169 attempts.
Found shorter road: 7019 meters in 77 attempts.
Found shorter road: 7018 meters in 128 attempts.
Found shorter road: 7013 meters in 356 attempts.
Found shorter road: 7012 meters in 28 attempts.
Found shorter road: 7011 meters in 66 attempts.
Found shorter road: 7006 meters in 5 attempts.
Found shorter road: 7000 meters in 373 attempts.
Found shorter road: 6978 meters in 2 attempts.
Found shorter road: 6973 meters in 29 attempts.
Found shorter road: 6972 meters in 109 attempts.
Found shorter road: 6967 meters in 7 attempts.
Found shorter road: 6955 meters in 167 attempts.
```

5. Wyjaśnienie dziania algorytmu poszukującego najkrótszej ścieżki

Podczas poszukiwania najkrótszej ścieżki sprawdzana była ilość prób od momentu poprzednio znalezionej krótszej ścieżki.

Jeśli wartość się nie zmieniła od po wykonaniu 10000 próbnym losowań kolejnej ścieżki, program kończy swoje działanie.

Logi z działania programu zostały zapisane w pliku tekstowym, który jest przechowywany do momentu uruchomienia go ponownie. Znajduje się on również w ścieżce gdzie jest zapisana macierz [N x N] oraz plik konfiguracyjny lecz w pliku `log.log`.

6. Ostatnim już krokiem jest wyświetlenie rezultatu działania programu

Rezultat wyświetlany na konsoli zawiera informacje:

- Wartość minimalna i maksymalna pomiędzy miastami wprowadzona przez użytkownika podczas startu programu
- Liczba miast do zwiedzenia przez komiwojażera
- Długość pierwszej wyznaczonej trasy
- Długość ostatniej wyznaczonej trasy (prawdopodobnie najkrótsza, lecz nie mamy 100% pewności)

Informacje o trasach najkrótszej i najdłuższej można znaleźć w pliku `output.txt` wraz z innymi plikami wynikowymi.

- Łączna liczba wykonanych prób odnalezienia najkrótszej trasy
- Czas działania programu, czas poszukiwania najoptymalniejszej ścieżki

```
Found shorter road: 7006 meters in 5 attempts.  
Found shorter road: 7000 meters in 373 attempts.  
Found shorter road: 6978 meters in 2 attempts.  
Found shorter road: 6973 meters in 29 attempts.  
Found shorter road: 6972 meters in 109 attempts.  
Found shorter road: 6967 meters in 7 attempts.  
Found shorter road: 6955 meters in 167 attempts.  
  
Summary result:  
Road length between cities: [10 - 99]  
Number of cities to visit: 500  
First road: Road length: 27269 through 500 cities.  
Last road: Road length: 6955 through 500 cities.  
Total number of attempts to find the shortest road: 7375333  
Program has ended working after: 103 seconds.
```

7. Użytkownik wcisnął dowolny klawisz. Program kończy działanie.