CS5592- Project Report

Kholood Alhejori: khaybf 16234808
Nouf Alrasheed: nanxb 16238212

Emergency Vehicle Dispatching System

Fall -2017

Project Description:

Emergency Vehicle Dispatching System:

We applied the dispatching system on 48 ZipCodes with 82 connections between them. Our system serves the user requests based on vehicles availability.

We import the inputs from .csv files that contain the needed information.

The basic system scenario:

1- Ask the user to specify his location by asking for the ZipCode.

2- print list of options for the user to choose the desired Emergency Vehicle.

3- after collecting the request from the user, check if the user location can provide the needed vehicles or not. If not, then help will be asked from the nearest ZipCode by Applying Dijkstra Algorithm to calculate the distances between the user location as a source node and the other zipCodes as the other vertices.

4. then pick the needed vehicles from the nearest locations and send them to the user.

5- print all the information in a text file for the user.

Our System Details:

Programming Language: C#.

Programming IDE: Visual Studio.

Input: Data.csv , Edge.csv , Node.csv

Classes Description:

1. **Vertex Class:**
   This class used to represent Vertex/Node in a graph.
   Contains (AddEdge) method to add a link between vertices.
2. **Edge Class:**
   To represent a link to a destination vertex.
   Graph Class:
   The last data structure class is Graph. Objects of this type hold entire graphs including vertices and edges.
3. **Dijkstra Class:**
   After setting up the needed data structures. Dijkstra Algorithm could be applied.
   Has three main functions:
   LoadGraph: which is the first step to set all of the node distances except the starting point to infinity.
   CalculateDijkstra: contains the code that traverses the graph until all vertices processed or only unreachable vertices remain. A "queue" of processed vertices is used. Originally, this contains all of the vertices. As vertices processed, they are removed from the collection.

RetrieveDistances: creates the output results in a dictionary that can be used by the calling method without awareness of the other classes.

4. **ZipCode and Info classes:**
   These classes were used to store data. Importantly, every ZipCode holds a collection of his Info(Vehicle Types).

5. **InfoRepository**:
   This class is used for data access. The central dictionary is of type IDictionary<string, Student> with the ZipCodes name as key and hidden inside the repository.

6. **Request Class:**
   Is the class where the user interacts with the system, and we use it to collect the needed information so we can process them.

Used Methodology:

Dijkstra Algorithm:

Dijkstra's algorithm has five steps that calculate the distance of each vertex in a graph. The steps are:

1. build the graph by setting the cost for each vertex to infinity, except for the source vertex, which has a cost of zero. Mark every vertex in the graph as unprocessed.
2. Choose the current vertex. This is the unprocessed vertex with the smallest, non-infinite cost. the algorithm has completed if every vertex in the graph has been processed, or only vertex with an infinite cost remains unprocessed.
3. Calculate the cost for all of the current vertex's unprocessed neighbors by adding the current cost to the length of the edge to the neighbor. For each neighbor, if the calculated cost is smaller than the neighbor's current cost, set the cost to the new value.
4. Mark the current vertex as processed.
5. Repeat the process from step 2.

Initialize the cost of each node to ∞

Initialize the cost of the source to 0

While there are unknown nodes left in the graph

  Select the unknown node with the lowest cost: n

  Mark n as known

  For each node a which is adjacent to n

    a's cost = min(a's old cost, n's cost + cost of (n, a))

- **Efficiency of Algorithms:**
  Priority queue then the complexity is O(|E|+|V^2 |)

➢ Assumptions (Use case):

- **Use case 1:**
  Request: single type at a time for single vehicle.

```
Please Enter Your Zip Code: 66070

 Enter 1 for Ambulance
 Enter 2 for Fire Truck
 Enter 3 for Police Car
 Enter 0 to quit
Please Enter Request Type : 2
How many veihcle you need:      1

 Enter 1 for Ambulance
 Enter 2 for Fire Truck
 Enter 3 for Police Car
 Enter 0 to quit
Please Enter Request Type : 0
Plrase wait, Your Request will processed
Your Request is
Ambulance:0
Fire Truck: 1
Police Car:  0
ZipCode :  66070  Ambulance :  10   FireTruck :  6  Policecar :  6

You Requested: 1  FireTruck(s)
FireTruck #1    Your VehicalID  is:  97
```

Request1

```
Your Request is
Ambulance:0
Fire Truck: 1
Police Car:  0
Your current ZipCode Information:
ZipCode :  66070  Ambulance :  10   FireTruck :  6  Policecar :  6

You Requested: 1  FireTruck(s)

FireTruck #1    Your VehicalID  is:  97


Thank you for using our system!
```

- **Use case 2:**
  Request: multi type at a time for single vehicle.

```
noufalrasheed — Visual Studio External Console — -bash — 80×44
Please Enter Your Zip Code: 66101

 Enter 1 for Ambulance
 Enter 2 for Fire Truck
 Enter 3 for Police Car
 Enter 0 to quit
 Please Enter Request Type : 1
How many veihcle you need:       1

 Enter 1 for Ambulance
 Enter 2 for Fire Truck
 Enter 3 for Police Car
 Enter 0 to quit
 Please Enter Request Type : 2
How many veihcle you need:       1

 Enter 1 for Ambulance
 Enter 2 for Fire Truck
 Enter 3 for Police Car
 Enter 0 to quit
 Please Enter Request Type : 3
How many veihcle you need:       1

 Enter 1 for Ambulance
 Enter 2 for Fire Truck
 Enter 3 for Police Car
 Enter 0 to quit
 Please Enter Request Type : 0
Plrase wait, Your Request will processed
Your Request is
Ambulance:1
Fire Truck: 1
Police Car:  1
ZipCode :  66101  Ambulance :  9   FireTruck :  7  Policecar :  14

You Requested: 1  Ambulance(s)
Ambulance #1     Your VehicalID  is:  17

You Requested: 1  FireTruck(s)
FireTruck #1     Your VehicalID  is:  83

You Requested: 1  PoliceCar(s)
PoliceCar #1     Your VehicalID  is:  88
```

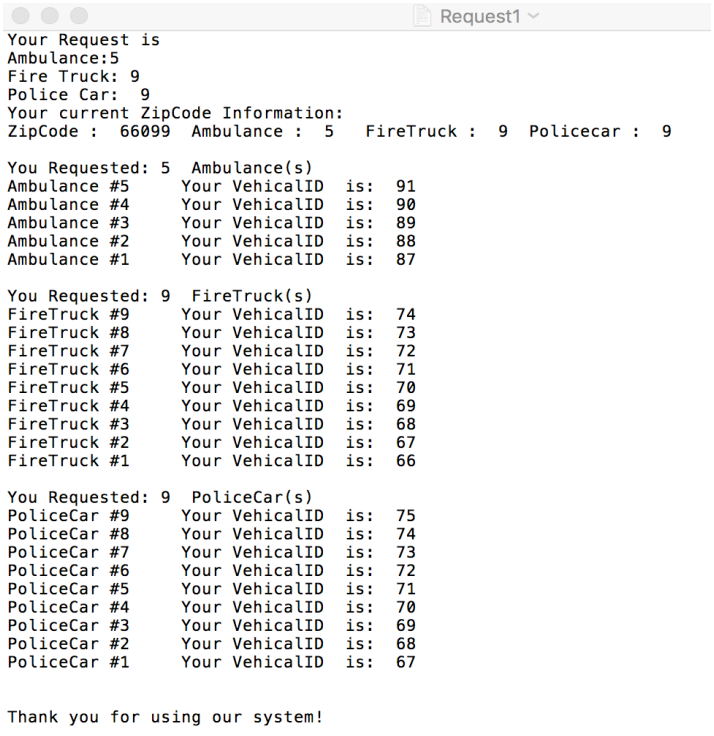
```
Request1
Your Request is
Ambulance:1
Fire Truck: 1
Police Car:  1
Your current ZipCode Information:
ZipCode :  66101  Ambulance :  9   FireTruck :  7  Policecar :  14

You Requested: 1  Ambulance(s)
Ambulance #1     Your VehicalID  is:  17

You Requested: 1  FireTruck(s)

FireTruck #1     Your VehicalID  is:  83

You Requested: 1  PoliceCar(s)

PoliceCar #1     Your VehicalID  is:  88


Thank you for using our system!
```

- **Use case 3:**
  Request: single type at a time for multi vehicle.

```
noufalrasheed — Visual Studio External Console — -bash — 80×34
Please Enter Your Zip Code: 66075

 Enter 1 for Ambulance
 Enter 2 for Fire Truck
 Enter 3 for Police Car
 Enter 0 to quit
 Please Enter Request Type : 3
How many veihcle you need:       11

 Enter 1 for Ambulance
 Enter 2 for Fire Truck
 Enter 3 for Police Car
 Enter 0 to quit
 Please Enter Request Type : 0
Plrase wait, Your Request will processed
Your Request is
Ambulance:0
Fire Truck: 0
Police Car:  11
ZipCode :  66075  Ambulance :  9   FireTruck :  7  Policecar :  11

You Requested: 11  PoliceCar(s)
PoliceCar #11    Your VehicalID  is:  69
PoliceCar #10    Your VehicalID  is:  68
PoliceCar #9     Your VehicalID  is:  67
PoliceCar #8     Your VehicalID  is:  66
PoliceCar #7     Your VehicalID  is:  65
PoliceCar #6     Your VehicalID  is:  64
PoliceCar #5     Your VehicalID  is:  63
PoliceCar #4     Your VehicalID  is:  62
PoliceCar #3     Your VehicalID  is:  61
PoliceCar #2     Your VehicalID  is:  60
PoliceCar #1     Your VehicalID  is:  59
```

```
Request1
Your Request is
Ambulance:0
Fire Truck: 0
Police Car:  11
Your current ZipCode Information:
ZipCode :  66075  Ambulance :  9   FireTruck :  7  Policecar :  11

You Requested: 11  PoliceCar(s)

PoliceCar #11    Your VehicalID  is:  69

PoliceCar #10    Your VehicalID  is:  68

PoliceCar #9     Your VehicalID  is:  67

PoliceCar #8     Your VehicalID  is:  66

PoliceCar #7     Your VehicalID  is:  65

PoliceCar #6     Your VehicalID  is:  64

PoliceCar #5     Your VehicalID  is:  63

PoliceCar #4     Your VehicalID  is:  62

PoliceCar #3     Your VehicalID  is:  61

PoliceCar #2     Your VehicalID  is:  60

PoliceCar #1     Your VehicalID  is:  59


Thank you for using our system!
```

- **Use case 4:**
  Request: multi type at a time for multi vehicle all available at same zip code.

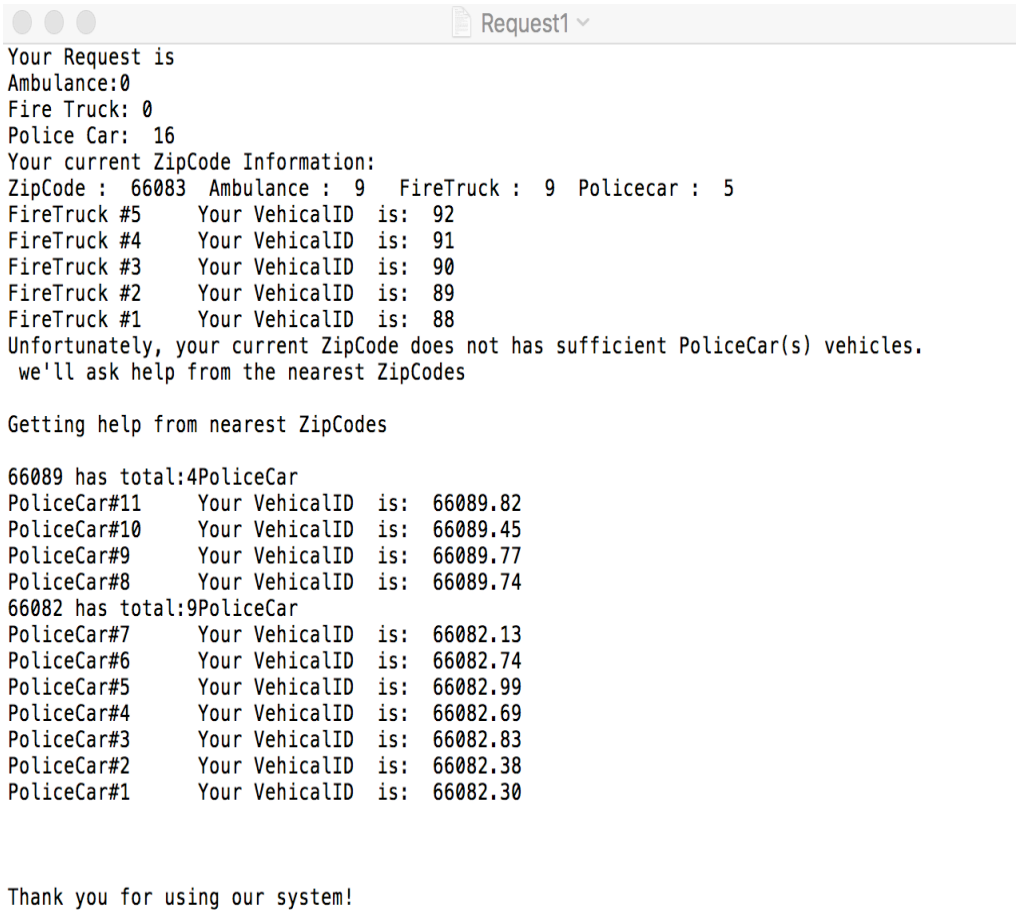


```
noufalrasheed — Visual Studio External Console — -bash — 80×35
Please Enter Your Zip Code: 66099

 Enter 1 for Ambulance
 Enter 2 for Fire Truck
 Enter 3 for Police Car
 Enter 0 to quit
 Please Enter Request Type : 1
How many veihcle you need:       4

 Enter 1 for Ambulance
 Enter 2 for Fire Truck
 Enter 3 for Police Car
 Enter 0 to quit
 Please Enter Request Type : 2
How many veihcle you need:       5

 Enter 1 for Ambulance
 Enter 2 for Fire Truck
 Enter 3 for Police Car
 Enter 0 to quit
 Please Enter Request Type : 3
How many veihcle you need:       9

 Enter 1 for Ambulance
 Enter 2 for Fire Truck
 Enter 3 for Police Car
 Enter 0 to quit
 Please Enter Request Type : 0
Plrase wait, Your Request will processed
Your Request is
Ambulance:4
Fire Truck: 5
Police Car:  9
ZipCode :  66099  Ambulance :  5   FireTruck :  9  Policecar :  9
```

```
noufalrasheed — Visual Studio External Console — -bash — 80×27
ZipCode :  66099  Ambulance :  5   FireTruck :  9  Policecar :  9

You Requested: 4  Ambulance(s)
Ambulance #4     Your VehicalID  is:  46
Ambulance #3     Your VehicalID  is:  45
Ambulance #2     Your VehicalID  is:  44
Ambulance #1     Your VehicalID  is:  43

You Requested: 5  FireTruck(s)
FireTruck #5     Your VehicalID  is:  42
FireTruck #4     Your VehicalID  is:  41
FireTruck #3     Your VehicalID  is:  40
FireTruck #2     Your VehicalID  is:  39
FireTruck #1     Your VehicalID  is:  38

You Requested: 9  PoliceCar(s)
PoliceCar #9     Your VehicalID  is:  17
PoliceCar #8     Your VehicalID  is:  16
PoliceCar #7     Your VehicalID  is:  15
PoliceCar #6     Your VehicalID  is:  14
PoliceCar #5     Your VehicalID  is:  13
PoliceCar #4     Your VehicalID  is:  12
PoliceCar #3     Your VehicalID  is:  11
PoliceCar #2     Your VehicalID  is:  10
PoliceCar #1     Your VehicalID  is:  9
```

Use case 5:

Request: single type at a time for multi vehicle that not available in the same zip code.

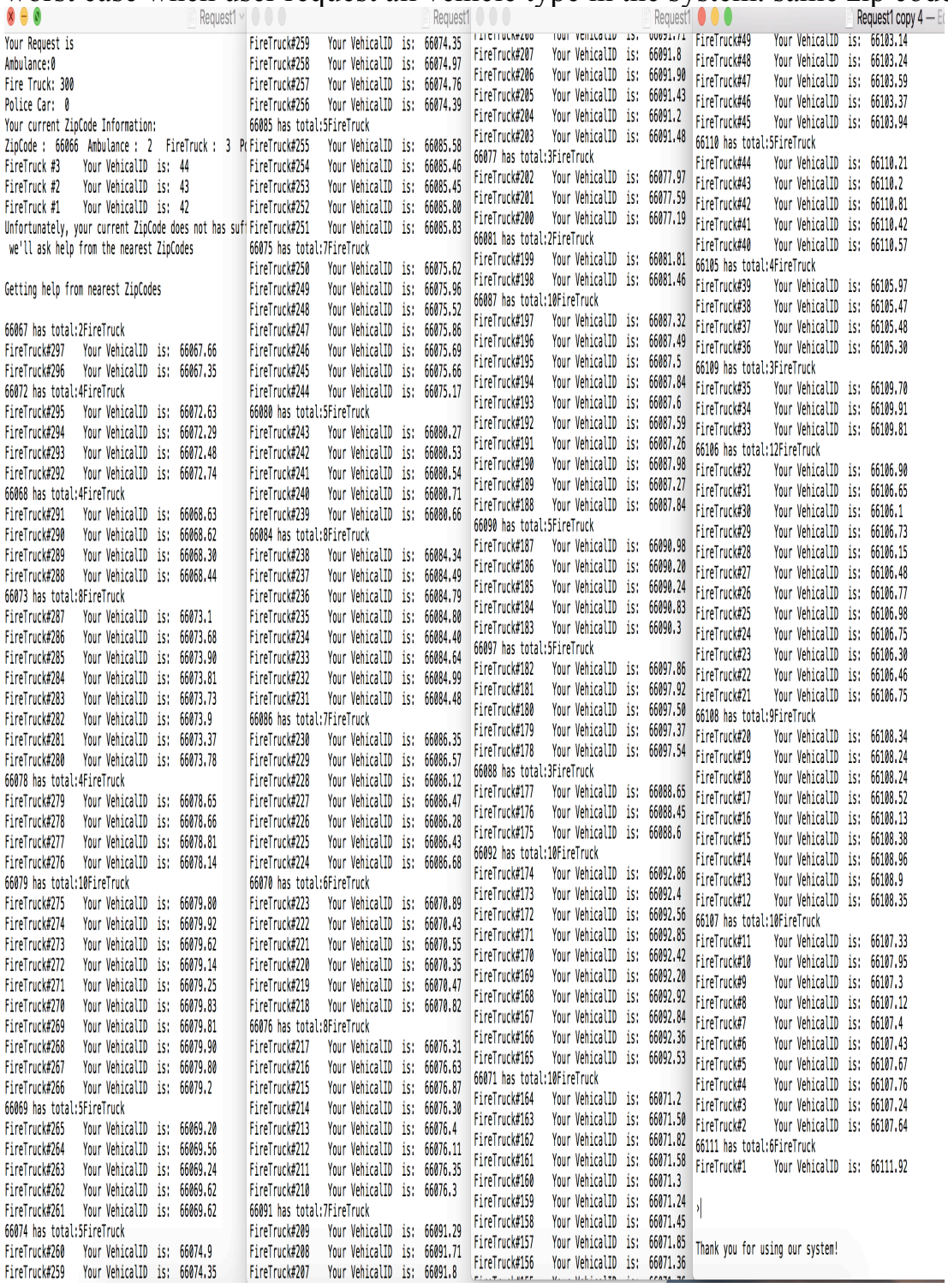


- **Use case 6:**
  Request: multi type at a time for multi vehicle all type not available at same zip code.

Request1

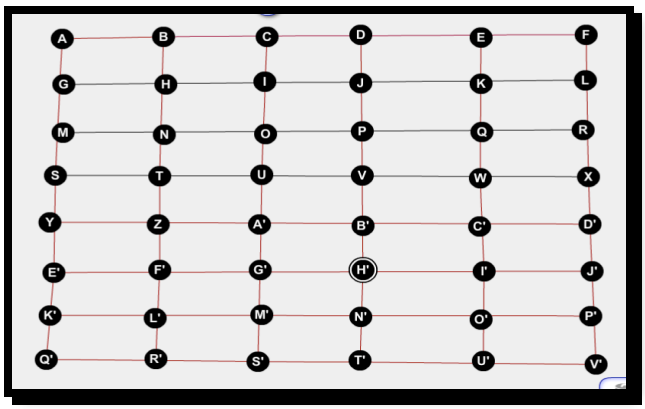
```
Your Request is
Ambulance:9
Fire Truck: 10
Police Car:  10
Your current ZipCode Information:
ZipCode :  66105  Ambulance :  4   FireTruck :  4  Policecar :  8
Ambulance #4     Your VehicalID  is:  19
Ambulance #3     Your VehicalID  is:  18
Ambulance #2     Your VehicalID  is:  17
Ambulance #1     Your VehicalID  is:  16
Unfortunately, your current ZipCode does not has sufficient Ambulance vehicle(s).
 we'll ask help from the nearest ZipCodes

FireTruck #4     Your VehicalID  is:  28
FireTruck #3     Your VehicalID  is:  27
FireTruck #2     Your VehicalID  is:  26
FireTruck #1     Your VehicalID  is:  25
Unfortunately, your current ZipCode does not has sufficient FireTruck(s) vehicles.
 we'll ask help from the nearest ZipCodes

FireTruck #8     Your VehicalID  is:  13
FireTruck #7     Your VehicalID  is:  12
FireTruck #6     Your VehicalID  is:  11
FireTruck #5     Your VehicalID  is:  10
FireTruck #4     Your VehicalID  is:  9
FireTruck #3     Your VehicalID  is:  8
FireTruck #2     Your VehicalID  is:  7
FireTruck #1     Your VehicalID  is:  6
Unfortunately, your current ZipCode does not has sufficient PoliceCar(s) vehicles.
 we'll ask help from the nearest ZipCodes

Getting help from nearest ZipCodes

66106 has total:11Ambulance
Ambulance#5      Your VehicalID  is:  66106.86
Ambulance#4      Your VehicalID  is:  66106.17
Ambulance#3      Your VehicalID  is:  66106.79
Ambulance#2      Your VehicalID  is:  66106.29
Ambulance#1      Your VehicalID  is:  66106.20

Getting help from nearest ZipCodes

66106 has total:12FireTruck
FireTruck#6      Your VehicalID  is:  66106.39
FireTruck#5      Your VehicalID  is:  66106.46
FireTruck#4      Your VehicalID  is:  66106.50
FireTruck#3      Your VehicalID  is:  66106.51
FireTruck#2      Your VehicalID  is:  66106.65
FireTruck#1      Your VehicalID  is:  66106.22

Getting help from nearest ZipCodes

66106 has total:8PoliceCar
PoliceCar#2      Your VehicalID  is:  66106.39
PoliceCar#1      Your VehicalID  is:  66106.46


Thank you for using our system!
```

- **Use case 7:**
  worst case when user request all vehicle type in the system. same zip code.



➢ Data:

In our system we make 48 nodes as zip code and 82 edges



	Source	Zipcod	Cost
2	66066	66067	6
3	66067	66068	5
4	66068	66069	8
5	66069	66070	7
6	66070	66071	9
7	66071	66077	3
8	66070	66076	2
9	66069	66075	7
10	66068	66074	12
11	66067	66073	9
12	66066	66072	6
13	66072	66073	6
14	66073	66074	7
15	66074	66075	3
16	66075	66076	4
17	66076	66077	4
18	66077	66083	12
19	66076	66082	10
20	66075	66081	9
21	66074	66080	3

This table represent the edges information(source,destination,cost)

	ZipCode	Ambulance	FireTruck	Policecar
2	66066	2	3	10
3	66067	12	2	9
4	66068	5	4	4
5	66069	14	5	5
6	66070	10	6	6
7	66071	9	10	7
8	66072	20	4	9
9	66073	5	8	3
10	66074	10	5	5
11	66075	9	7	11
12	66076	6	8	12
13	66077	11	3	14
14	66078	8	4	4
15	66079	3	10	5
16	66080	5	5	6
17	66081	12	2	7
18	66082	10	3	9
19	66083	9	9	5
20	66084	5	8	4
21	66085	3	5	9

This table represent the availability information for each type

GitHub:
https://github.com/noufcs/AlgorithmProject

References:

Books:

data structures and algorithms using c# michael mcmillan

Website:

http://www.marcinkossakowski.com/finding-shortest-path-using-dijkstras-algorithm/
https://en.wikipedia.org/wiki/Dijkstra%27s_algorithm
http://www.geeksforgeeks.org/greedy-algorithms-set-6-dijkstras-shortest-path-algorithm/