DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Shahbad Daulatpur Delhi-110042

SUBMITTED AGAINST THE MTE COMPONENT OF THE SUBJECT

Algorithm Design and Analysis (CS262)

DEPARTMENT OF APPLIED MATHEMATICS

"FOOD DELIVERY SYSTEM"

Submitted to:

Ms Neha Punetha

Submitted by:

Kashan Akram 2K20/MC/71

DECLARATION

I hereby declare that this project report entitled "**FOOD DELIVERY SYSTEM**" is done by us and is my own effort and that no part has been plagiarized without citations. The matter embodied in this project work has not been submitted earlier for the award of any degree or diploma to the best of my knowledge and belief.

Kashan Akram 2K20/MC/71

INDEX

S. No	TOPIC
1.	ACKNOWLEDGEMENT
2.	MOTIVATION
3.	OBJECTIVE
4.	EXPERIMENTAL DESIGN
5.	EXPLANATION
6.	TIME AND SPACE COMPLEXITIES
7.	FULL CODE
8.	MODULE DETAILS
9.	CONCLUSION
10.	CONTRIBUTION
11.	REFERENCES

ACKNOWLEDGEMENT

An institutional project is a golden opportunity for learning and self-development. I consider my self very lucky and honoured to have **Ms Neha Punetha**, Professor in Applied Mathematics department, Delhi Technological University, a wonderful teacher who led us through in completion of this project.

I wish to express my indebted gratitude and special thanks to **Ms Neha Punetha**, Professor in applied mathematics department, Delhi Technological University, my mentor who in spite of being extraordinarily busy with her engagements in various projects and classes, took time out to hear, guide and keep us on the correct path and allowing us to carry out my Project work. I couldn't have completed my project without her support and guidance.

I choose this moment to acknowledge their contribution gratefully and offer them my sincere gratitude.

A humble "Thank you'' to you, ma'am.

Kashan Akram 2K20/MC/71

MOTIVATION

I have always craved for good food during the night. During a late-night study session too far from home, I need someone to bring me SOME COOKED FOOD !!, or I'd have to continue surviving on Maggi. Now, these apps primarily maintain two large segments:

- One is the process of ordering.
- The other is the process of bringing their product to the customers.

and both these are one hefty job. These require large Tech support, a big chain of food restaurants, a large amount of people working to deliver food. These provide big employment opportunities for all segments. I tried and loved creating a working project of my own with similar features.

OBJECTIVE

- The goal was to design an efficient food delivery system for students living in and around DTU.
- To provide best food in budget ranging from street food to an expensive restaurant.
- It also gives us a scope to expand this idea throughout the city.
- Using efficient data structures, I overcame the problem of managing food menus for different restaurants.
- I implemented my knowledge of graph theory to design a map around the college giving students a wide variety of restaurants to choose from.

EXPERIMENTAL DESIGN

1. Data Structures used:

- **Arrays and vectors:** A vector are a sequence container class that implements dynamic array, means size automatically changes when appending elements. A vector stores the elements in contiguous memory locations and allocates the memory as needed at run time.
- **HashMap:** HashMap is a Map based collection class that is used for storing Key & value pairs, in C++, a hash map is known as an unordered map <key, value>. Hash maps are implemented using hash functions, a key is processed by a hash function and the returned value gives us the index for the value.
- **Set:** Sets are a type of associative containers in which each element has to be unique, because the value of the element identifies it. A set is implemented using an AVL tree. AVL
- **Trees (Adelson, Velski & Landis):** a balancing BST (Binary Search Tree), being balanced, insertion, search and deletion operations are done in O(log n) time complexity.
- **Graphs:** Formally, a graph is a pair of sets (V, E), where V is the set of vertices and E is the set of edges, connecting the pairs of vertices. I have implemented a weighted undirected graph to denote two-way roads.

2. Algorithms and Concepts used:

- **Greedy Method:** A greedy algorithm is a simple, intuitive algorithm that is used in optimization problems. The algorithm makes the optimal choice at each step as it attempts to find the overall optimal way to solve the entire problem.
- **Dijkstra algorithm:** Given a graph and a source vertex in the graph, find shortest paths from source to all vertices in the given graph. I generate a *SPT (shortest path tree)* with given source as root. I maintain two sets, one set contains vertices included in shortest path tree, other set includes vertices not yet included in shortest path tree. At every step of the algorithm, I find a vertex which is in the other set (set of not yet included) and has a minimum distance from the source.
- **Path printing using Dijkstra algorithm:** I maintain parent node for every node, parent of a node is the node by which current node is explored in the Dijkstra algorithm.
- **Multi-source Dijkstra:** To get shortest distance of a place to multiple sources.
- **Sorting and Searching** (binary search, linear search)
- Elementary concepts like loops, pointers, pairs, recursion, memoisation

EXPLANATION

Created a data set of restaurants and their food items along with the distance from the selected destination, and created a graph with nodes as all the restaurants and the locations from the data set with weighted edges the distance between the two nodes.

Using greedy paradigm and Dijkstra algorithm the shortest path was calculated to simulate a delivery system for selected nodes.

The user may choose the location of their whereabouts and according to the location a list of top restaurants will be displayed. The user will then be asked to choose a filter to display the restaurants on the basis of the price or ETA from your location. The user may then select the restaurant and further the food items they want to order by inputting serial number and quantity afterwards there's an option to use discount coupons and all.

The mode of payment will also be asked by the module and finally your food will be order successfully and module will show a "map" of all the points your delivery agent is going to traverse through.

These points are found using Dijkstra's algorithm according to the nodes I selected.

COMPLEXITIES OF THE ALGORITHMS USED

1. DIJKSTRA ALGORITHM:

- The Time Complexity of Dijkstra's Algorithm is O(V^2)
- The Space Complexity of Dijkstra's Algorithm is O(V^2)
- The Time Complexity of multi-source Dijkstra's Algorithm is O(V^2)
- The Space Complexity of multi-source Dijkstra's Algorithm is O(V^2)

2. GREEDY ALGORITHMS:

- The Time Complexity of Greedy Method for finding the path with maximum reward is O(m*n) or more generally O(n^2).
- The Space Complexity of Greedy Method for finding the path with maximum reward is O(m*n) or more generally O(n^2).

3. SORTING ALGORITHMS:

Sort	Best case time	Average case time	Worst case time	Space
Bubble sort	Ω(n)	Θ(n^2)	O(n^2)	O(1)
Selection sort	Ω(n^2)	Θ(n^2)	O(n^2)	O(1)
Insertion sort	Ω(n)	Θ(n^2)	O(n^2)	O(1)
Merge sort	Ω(nlogn)	Θ(nlogn)	O(nlogn)	O(1)
Heap sort	Ω(nlogn)	Θ(nlogn)	O(nlogn)	O(1)
Quick sort	Ω(nlogn)	Θ(nlogn)	O(n^2)	O(nlogn)

4. SEARCHING ALGORITHMS:

Search	Best Case time	Worst case time	Space
Linear Search	O(1)	O(n)	O(1)
Binary search	O(1)	O(logn)	O(1)

FULL CODE

```
#include<bits/stdc++.h>
using namespace std;
#define umap unordered_map
#define mpp(x,y) make_pair(x,y)
#define pb push_back
#define ff first
#define ss second
umap<int,string> places;
umap<string,int> revPlaces;
umap<int,vector<pair<int,int>>> graph;
vector<pair<int,int>> foodplaces;
umap<int,int> par;
umap<int,int> dist;
vector<int> placesOfStay;
void printPath(int rest);
void addEdges();
void markings();
void dijkstra(int rest);
void addMenus();
umap<int,set<pair<string,int>>> menus;
vector<string> coupons={"DTU_BEST","SAVE25","BAWANABOIS","HOSTEL25"};
// struct comparator
// {
// bool operator()(const pair<string,int> &a,const pair<string,int> &b)const{
// return a.ss<b.ss;
// }
// };
bool comp1(pair<pair<int,int>,int> &a,pair<pair<int,int>,int> &b)
```

```
{
 return a.ff.ss<b.ff.ss;
}
bool comp2(pair<pair<int,int>,int> &a,pair<pair<int,int>,int> &b)
{
 return a.ss<b.ss;
}
#define all(x) x.begin(),x.end()
void printmenu(int rest);
int main()
{

 markings();
 addEdges();
 addMenus();
 cout<<"\t\t\t\t FOODinTHRIVE "<<endl;
 cout<<"\tWhere are you currently staying at?"<<endl;
 for(int i=0;i<placesOfStay.size();i++)
 {
 cout<<"\t"<<i+1<<". "<<places[placesOfStay[i]]<<endl;
 }
 cout<<endl;
 cout<<"\t";
 int r;
 cin>>r;
 r--;
 int home=placesOfStay[r];
 cout<<"\tTop Picks for you"<<endl;
 dijkstra(home);
 cout<<"\tList of Restaurants: "<<endl;
 for(int i=0;i<foodplaces.size();i++)
```

```
{
cout<<"\t"<<i+1<<". "<<places[foodplaces[i].ff]<<endl;
}
cout<<endl;
int x;
cout<<"\t\tFilter according to:\n\t1. Price \n\t2. ETA from your place\n";
cout<<"\t";
cin>>x;

vector<pair<pair<int,int>,int>> v(foodplaces.size());
for(int i=0;i<foodplaces.size();i++)
{
v[i].ff.ff=foodplaces[i].ff;
v[i].ff.ss=foodplaces[i].ss;
v[i].ss=dist[foodplaces[i].ff];
}
if(x==1)
{
cout<<"\t\tSorted from LoIst to Highest Price\n\n";

sort(all(v),comp1);
for(int i=0;i<v.size();i++)
{
cout<<"\t";
cout.width(3);cout<<left<<i+1;
cout<<". ";
cout.width(50);cout<<left<<places[v[i].ff.ff]<<"\t";
cout.width(4);cout<<left<<"Rs"<<v[i].ff.ss<<"\t ";
cout.width(5);cout<<left<<v[i].ss<<"mins\n";
}
cout<<"\n\n";;
```

```
}
else
{
cout<<"\t\tSorted from LoIst to Highest ETA\n\n";

sort(all(v),comp2);
for(int i=0;i<v.size();i++)
{
cout<<"\t";
cout.width(3);cout<<left<<i+1;
cout<<". ";
cout.width(50);cout<<left<<places[v[i].ff.ff]<<"\t";
cout.width(4);cout<<left<<"Rs"<<v[i].ff.ss<<"\t ";
cout.width(5);cout<<left<<v[i].ss<<"mins\n";
}
cout<<"\n\n";
}
cout<<"\tTell us your choice of Restaurant\n";
cout<<"\t";
cin>>x;
x--;
int rest=v[x].ff.ff;

cout<<"\t\tMenu for "<<places[rest]<<endl;
printmenu(rest);
x=1;
cout<<"\t\tHow would you like to make the payment\n\n";
cout<<"\t";
cout<<x++<<". ";
cout<<"Cash\n";
cout<<"\t";
```

```
cout<<x++<<". ";
cout<<"Credit/Debit Card\n";
cout<<"\t";
cout<<x++<<". ";
cout<<"Paytm / PhonePay / GooglePay\n";
cout<<"\t";
cin>>x;
if(x==1)
{

}
else if(x==2)
{
long long int a,b;
cout<<"\t";
cout<<left<<"Enter card number and CVV\n";
cout<<"\t";
cin>>a>>b;

}
else if(x==3)
{
long long int a;
cout<<"\t";
cout<<left<<"Enter Phone number\n\n";
cout<<"\t";
cin>>a;
}

cout<<"\t";
cout<<left<<"Food is Preparing\n\n";
```

```
 cout<<"\t";
 cout<<left<<"Our service will be at your doorstep in Expected Time(ETA): "<<dist
[rest]<<" minutes"<<"\n";
 cout<<"\tOur agent is coming via: \n";
 printPath(rest);



 return 0;
}
bool f(string a)
{
 for(int i=0;i<coupons.size();i++)
 if(coupons[i]==a)
 return true;
 return false;
}
void printmenu(int rest)
{
 set<pair<string,int>>s;

 s=menus[rest];
 vector<pair<string,int>> v(s.begin(),s.end());
 int i=0;
 for(auto it=s.begin();it!=s.end();it++)
 {
 cout<<"\t";
 cout.width(3);cout<<left<<++i<<". ";
 cout.width(50);cout<<left<<(*it).ff;
 cout<<" "<<(*it).ss<<"\n";
 }
 cout<<"\n\n";
```

```
int price=0;
set<pair<string,int>> currentMenu;
cout<<"\tEnter the number of items you want to add\n\t";
int n,x,qty;
cout<<"\t";
cin>>n;
for(int i=0;i<n;i++)
{
cout<<"\tEnter serial number and quantity\n\t";
cin>>x>>qty;
x--;
int pr=qty * v[x].ss;
price+=pr;
currentMenu.insert(mpp(v[i].ff,pr));
}
cout<<"\t\nYour final Menu looks like: \n\n";
i=0;
for(auto it=currentMenu.begin();it!=currentMenu.end();it++)
{
cout<<"\t";
cout.width(3);cout<<left<<++i<<". ";
cout.width(50);cout<<left<<(*it).ff;
cout<<" "<<(*it).ss<<"\n";
}
cout<<"\n";
cout<<"\t ";
cout.width(53);cout<<left<<"Final Price to pay: "<<price<<"\n\n";
cout<<"\tDo you have a discount coupon\n\t";
string a;
cin>>a;
int val=4;
```

```
 sort(all(coupons));
 while(val--){
 if(f(a))
 {
 cout<<"\t\t\t ***CONGRATS 25% discount applied***\n";
 cout.width(55);cout<<left<<"\tYour final price is: "<<(float)0.75*price<<"\n\n";
 break;
 }
 else if(val)
 {
 cout<<"\tOops coupon failed,Do you want to retry?\n\t1.Retry 0.Exit?\n\t";
 int ret;
 cin>>ret;
 if(!ret)
 {
 cout<<"\t\t"<<"Your final price is: "<<price<<"\n\n";
 }
 }
 else
 {
 cout<<"\t\tCoupons Failed :("<<"\tYour final price is: "<<price<<"\n\n";
 }
 }
}
void printPath(int cur)
{
 if(par[cur]==cur)
 {
 cout<<places[cur]<<endl;
 return;
 }
```

```
 cout<<places[cur]<<" -> ";
 printPath(par[cur]);
}
void dijkstra(int rest)
{
 int src=rest;
 for(auto ele:places)
 {
 dist[ele.ff]=INT_MAX;
 }
 dist[src]=0;
 par[src]=src;
 set<pair<int,int>> s; // <dist,place>
 s.insert(mpp(dist[src],src));
 while(!s.empty())
 {
 auto aage=s.begin();
 int node=(*aage).ss;
 int curdist=(*aage).ff;
 // cout<<node<<endl;
 // cout<<curdist<<endl;
 s.erase(aage);

 for(auto ele:graph[node])
 {
 int nbr=ele.ff;
 int edgewt=ele.ss;
 // cout<<places[nbr]<<" "<<dist[nbr]<<endl;
 int a=dist[node]+edgewt;
 if(dist[nbr]> a)
 {
```

```
 auto x=s.find(mpp(dist[nbr],nbr));
 if(x!=s.end())
 {
 s.erase(mpp(dist[nbr],nbr));
 s.insert(mpp(a,nbr));
 }
 else
 {
 s.insert(mpp(a,nbr));
 }
 dist[nbr]=a;
 par[nbr]=node;
 }
 }
 }
 // int i=0;
 // for(auto ele:dist)
 // {
 // cout<<++i;
 // cout<<setw(20)<<places[ele.ff]<<" "<<ele.ss<<endl;
 // }
}
void addMenus()
{
 set<pair<string,int>> s;
 //mc donalds
 s.insert(mpp("Chicken_Kebab_Burger",85));
 s.insert(mpp("Kebab_Double_Patty_Burger",129));
 s.insert(mpp("Maharaja_Mac",215));
 s.insert(mpp("O_Fish_Double_Patty_Burger",279));
 s.insert(mpp("Filet_O_Fish Burger",153));
```

```
s.insert(mpp("Grilled Schezwan Chicken Double Patty Burger",115));
s.insert(mpp("McAloo Tikki Burger",49));
s.insert(mpp("McAloo Tikki Double Patty Burger",74));
s.insert(mpp("McChicken Burger",124));
s.insert(mpp("Chicken McNuggets Piri Piri ( 20 pc)",360));
s.insert(mpp("Chicken McNuggets Piri Piri ( 6 pc)",175));
s.insert(mpp("Chicken McNuggets Piri Piri ( 9 pc)",212));
s.insert(mpp("Chicken McWings - 12 pcs",400));
s.insert(mpp("Chicken McWings - 4 pcs",140));
s.insert(mpp("Chicken McWings - 8 pcs",270));
s.insert(mpp("Fries (L)",116));
s.insert(mpp("Fries (M)",98));
s.insert(mpp("Fries (R)",60));
menus[21]=s;
s.clear();
// s=menus[21];
// int i=0;
// for(auto it=s.begin();it!=s.end();it++)
// {
// cout<<"\t";
// cout.width(3);cout<<left<<++i<<". ";
// cout.width(50);cout<<left<<(*it).ff;
// cout<<" "<<(*it).ss<<"\n";
// }
//the chef
s.insert(mpp("french fries",70));
s.insert(mpp("chilly potato",100));
s.insert(mpp("paneer tikka",100));
s.insert(mpp("paneer tikka roll",120));
s.insert(mpp("chicken tikka roll",140));
s.insert(mpp("rajma",8));
```

```
s.insert(mpp("chole",70));
s.insert(mpp("shahi paneer",170));
s.insert(mpp("kadhai paneer",180));
s.insert(mpp("dal makhani",160));
menus[6]=s;
s.clear();
//Aggarwal
s.insert(mpp("Agarwal_ Laddu",50));
s.insert(mpp("Badam_Soan_Papdi",170));
s.insert(mpp("Bombay_Jalebi",135));
s.insert(mpp("Kalakand",150));
s.insert(mpp("Badam_Halwa",285 ));
s.insert(mpp("Gulab_Jamun (1 Pc)",36));
s.insert(mpp("Rasgulla (1 Pc)",36));
s.insert(mpp("Rasmalai(1 Pc)",36));
s.insert(mpp("Kaju_Katli",260));
s.insert(mpp("Samosa",22));
s.insert(mpp("Panipuri",60));
menus[11]=s;
s.clear();
//bittu_tikki
s.insert(mpp("bittu_special_tikki",95));
s.insert(mpp("raj_kachori",110));
s.insert(mpp("plain_dhokla",55));
s.insert(mpp("pani_puri",50));
s.insert(mpp("BTW_special_thali",276));
s.insert(mpp("tandoori_platter",248));
s.insert(mpp("chinese_platter",219));
s.insert(mpp("sambhar_vada",70));
s.insert(mpp("shahi_paneer",225));
s.insert(mpp("dal_makhani",225));
```

```
menus[26]=s;
//KFC

s.insert(mpp("Zinger_Burger ",150.00));
s.insert(mpp("Zinger_Burger_Combo", 229.00));
s.insert(mpp("2 x Zinger_Burgers",219.00));
s.insert(mpp("5 in 1 Zinger_Burger_Box",239.00));
s.insert(mpp("KFC_Favorites_Meal (Chicken_Zinger, 4 Hot_Wings, Large_Popcorn)",
375.00));
s.insert(mpp("Veg_Zinger_Burger", 140.00));
s.insert(mpp("8PC Hot & Crispy" ,650.00));
s.insert(mpp("6PC Hot & Crispy" ,550.00));
s.insert(mpp("4PC Hot & Crispy." , 380.00));
s.insert(mpp("2PC Hot & Crispy" ,200.00));
s.insert(mpp("8PC Smoky Grilled",650.00));
s.insert(mpp("6PC Smoky Grilled", 550.00));
s.insert(mpp("4PC Smoky Grilled", 380.00));
s.insert(mpp("Popcorn Rice Combo", 219.00));
s.insert(mpp("Popcorn Rice Duo", 199.00));
s.insert(mpp("5 in 1 Rice Box", 239.00));
s.insert(mpp("Smoky Rice Bowl", 180.00));
s.insert(mpp("Smoky Rice Bowl & Pepsi" ,215.00));
s.insert(mpp("Chicken Rice Bowl", 135.00));
s.insert(mpp("Pepsi Can", 30.00));
s.insert(mpp("Pepsi Bottle", 60.00));
s.insert(mpp("Red Bull ",140.00));
menus[22]=s;
s.clear();
//bikaner
s.insert(mpp("shahi_paneer",260));
s.insert(mpp("dal_makhani",230));
```

```
s.insert(mpp("rajma_chawal",130));
s.insert(mpp("chole_chawal",130));
s.insert(mpp("kadhi_chawal",130 ));
s.insert(mpp("Deluxe_thali",290));
s.insert(mpp("plain_naan",50));
s.insert(mpp("butter_naan",70));
s.insert(mpp("lachha_parantha",60));
s.insert(mpp("paneer_tikka",240));
menus[16]=s;
s.clear();
// biryani_shop
s.insert(mpp("Awadhi_Veg_Paneer_Biryani",135));
s.insert(mpp("Hyderabadi_Kathal_Biryani",130));
s.insert(mpp("Awadhi_Egg_Biryani",125));
s.insert(mpp("Hyderabadi_Chicken_Biryani",185));
s.insert(mpp("Awadhi_Chicken_Biryani",165));
s.insert(mpp("Muradabadi_Chicken_Biryani",135));
s.insert(mpp("Butter_Chicken",185));
s.insert(mpp("Chicken_Korma",205));
s.insert(mpp("Purvanchali_Chicken_Masala",155));
s.insert(mpp("Purvanchali_Mutton_Masala",215));
s.insert(mpp("Kathal_Do_Pyaza",255));
s.insert(mpp("Paneer_Butter_Masala",315));
s.insert(mpp("Tawa_Roti",8));
s.insert(mpp("Ghee_Tawa_Roti",12));
s.insert(mpp("Butter_Tawa_Roti",12));
menus[7]=s;
s.clear();
//vaishno_dhaba
s.insert(mpp("Chilli_Paneer_Gravy",110));
s.insert(mpp("Kaju_Paneer",115));
```

```
s.insert(mpp("Makhana_Paneer",110));
s.insert(mpp("Aloo_Masala_Dry",55));
s.insert(mpp("Paneer_Bhurji",220));
s.insert(mpp("Kadai_Paneer",99));
s.insert(mpp("Paneer_Do_Pyaza",99));
s.insert(mpp("Paneer_Butter_Masala",99));
s.insert(mpp("Shahi_Paneer",99));
s.insert(mpp("Chole_Paneer",99));
s.insert(mpp("Veg_Kofta",55));
s.insert(mpp("Chole_Masala",55));
s.insert(mpp("Rajma_Masala",55));
s.insert(mpp("Dal_Makhani",60));
s.insert(mpp("Arhar_Dal_Makhani_Fry",55));
s.insert(mpp("Jeera_Rice",50));
s.insert(mpp("Sada_Chawal",45));
s.insert(mpp("Paneer_Fried_Rice",55));
s.insert(mpp("Tawa_Roti_Makhan_Se",12));
s.insert(mpp("Dahi_Vada",30));
s.insert(mpp("Boondi_Raita",20));
s.insert(mpp("Chole_Chawal",120));
s.insert(mpp("Rajma_Chawal",120));
s.insert(mpp("Dal_Chawal",120));
s.insert(mpp("Kadhi_Chawal",120));
s.insert(mpp("Kheer",35));
menus[12]=s;
s.clear();
//Apsara
s.insert(mpp("Rajma_Chawal",110));
s.insert(mpp("Chole_Chawal" ,110));
s.insert(mpp("Special_Thali",180));
s.insert(mpp("Deluxe_Thali",230));
```

```
s.insert(mpp("Veg_Cutlet",70));
s.insert(mpp("Veg_Spring_Roll",110));
s.insert(mpp("Chaap_Nuggets",90));
s.insert(mpp("French_Fries",80));
s.insert(mpp("Peri_Peri_Spicy_Fries",100));
s.insert(mpp("Gol_Gappe",25));
s.insert(mpp("Dahi_Bhalla",80));
s.insert(mpp("Raj_Kachori",100));
s.insert(mpp("Rajma",160));
s.insert(mpp("Mixed_Vegetable",160));
s.insert(mpp("Chana_Masala",160));
s.insert(mpp("Mattar_Paneer",170));
s.insert(mpp("Shahi_Paneer",180));
s.insert(mpp("Malai_Kofta",180));
s.insert(mpp("Kadai_Paneer",190));
s.insert(mpp("Paneer_Butter_Masala",190));
s.insert(mpp("Veg_Manchurian_Gravy",90));
s.insert(mpp("Dal_Makhani",160));
s.insert(mpp("Veg_Biryani",180));
s.insert(mpp("Veg_Rice",80));
s.insert(mpp("Butter_Jeera_Rice",110));
s.insert(mpp("Roti",14));
s.insert(mpp("Butter_Roti",18));
s.insert(mpp("Rumali_Roti",12));
s.insert(mpp("Missi_Roti",30));
s.insert(mpp("Plain_Naan",27));
s.insert(mpp("Butter_Naan",35));
s.insert(mpp("Stuffed_Paratha",50));
s.insert(mpp("Lassi",50));
menus[15]=s;
s.clear();
```

```
 //al_Qureshi
 s.insert(mpp("Tandoori_Chicken",140));
 s.insert(mpp("Afghani_Chicken",155));
 s.insert(mpp("Chicken_Fry",115));
 s.insert(mpp("Chicken_Lollipop",130));
 s.insert(mpp("Tawa_Chicken",220));
 s.insert(mpp("Butter_Chicken",220));
 s.insert(mpp("Chicken_Tikka(6 Pcs)",215));
 s.insert(mpp("Malai_Tikka(6 Pcs)",220));
 s.insert(mpp("Chicken_Seekh_Kebab(6 Pcs)",155));
 s.insert(mpp("Boneless_Tawa_Chicken",250));
 s.insert(mpp("Boneless_Butter_Chicken",250));
 s.insert(mpp("Paneer_Tikka",155));
 s.insert(mpp("Chicken_Tikka_Roll",150));
 s.insert(mpp("Chicken_Kebab_Roll",130));
 s.insert(mpp("Rumali_Roti",10));
 s.insert(mpp("Tandoori_Roti",15));
 s.insert(mpp("Butter_Naan",40));
 s.insert(mpp("Plain_Naan",25));
 menus[18]=s;
 s.clear();
}
void markings()
{
 places[1]="DTU";
 places[2]="DEVTA_PG";
 places[3]="B3_15";
 places[4]="GARG_PG";
 places[5]="FUTURE_POINT";
 places[6]="THE_CHEF";
 places[7]="BIRYANI_SHOP";
```

```
places[8]="TRIKHA_INSTITUTE";
places[9]="BAREJA_CHEMIST";
places[10]="DIGITAL_ZONE";
places[11]="AGGARWAL_SIETS";
places[12]="VAISHO_DHABA";
places[13]="DELHI_JAL_BOARD";
places[14]="PCR";
places[15]="APSARA_RESTAURANT";
places[16]="BIKANER";
places[17]="G3S";
places[18]="AL_QURESHI";
places[19]="MARUTI_TRUE_VALUE";
places[20]="SSCBS";
places[21]="MC_DONALDS";
places[22]="KFC_SEC12_MALL";
places[23]="ISHAAN_HOSPITAL";
places[24]="AMAR_COLONY";
places[25]="MASJID_UMAR";
places[26]="BITTU_TIKKI";
places[27]="SAMAYPUR_BADLI";
places[28]="HYUNDAI_CENTRE";
for(auto ele:places)
{
revPlaces[ele.ss]=ele.ff;
}
placesOfStay.push_back(1);
placesOfStay.push_back(2);
placesOfStay.push_back(3);
placesOfStay.push_back(4);
placesOfStay.push_back(27);
foodplaces.push_back(mpp(6,350));
```

```
 foodplaces.push_back(mpp(7,300));
 foodplaces.push_back(mpp(11,200));
 foodplaces.push_back(mpp(12,250));
 foodplaces.push_back(mpp(15,400));
 foodplaces.push_back(mpp(16,450));
 foodplaces.push_back(mpp(18,500));
 foodplaces.push_back(mpp(21,220));
 foodplaces.push_back(mpp(22,280));
 foodplaces.push_back(mpp(26,230));
}
void help(int a,int b,int wt)
{
 graph[a].push_back(mpp(b,wt));
 graph[b].push_back(mpp(a,wt));
}
void addEdges()
{
 help(1,2,1);
 help(5,2,2);
 help(5,6,1);
 help(1,6,2);
 help(1,5,1);
 help(1,7,5);
 help(6,7,2);
 help(7,8,3);
 help(8,9,4);
 help(10,6,2);
 help(9,4,3);
 help(10,4,2);
 help(9,11,2);
 help(4,11,1);
```

```
 help(11,12,4);
 help(11,13,1);
 help(14,13,2);
 help(3,14,0);
 help(13,15,4);
 help(13,16,3);
 help(16,15,6);
 help(15,17,4);
 help(16,17,4);
 help(18,16,1);
 help(8,9,4);
 help(19,16,7);
 help(20,17,6);
 help(20,21,4);
 help(20,22,3);
 help(22,23,2);
 help(21,23,2);
 help(24,21,1);
 help(23,25,2);
 help(20,26,2);
 help(25,19,10);
 help(26,27,11);
 help(27,28,3);
 help(25,28,6);

}
```

MODULE DETAILS

```
                        FOODinTHRIVE
Where are you currently staying at?
1. DTU
2. DEVTA_PG
3. B3_15
4. GARG_PG
5. SAMAYPUR_BADLI

1
```

```
Top Picks for you
List of Restaurants:
1. THE_CHEF
2. BIRYANI_SHOP
3. AGGARWAL_SWEETS
4. VAISHO_DHABA
5. APSARA_RESTAURANT
6. BIKANER
7. AL_QURESHI
8. MC_DONALDS
9. KFC_SEC12_MALL
10. BITTU_TIKKI
```

```
        Filter according to:
1. Price
2. ETA from your place
2
```

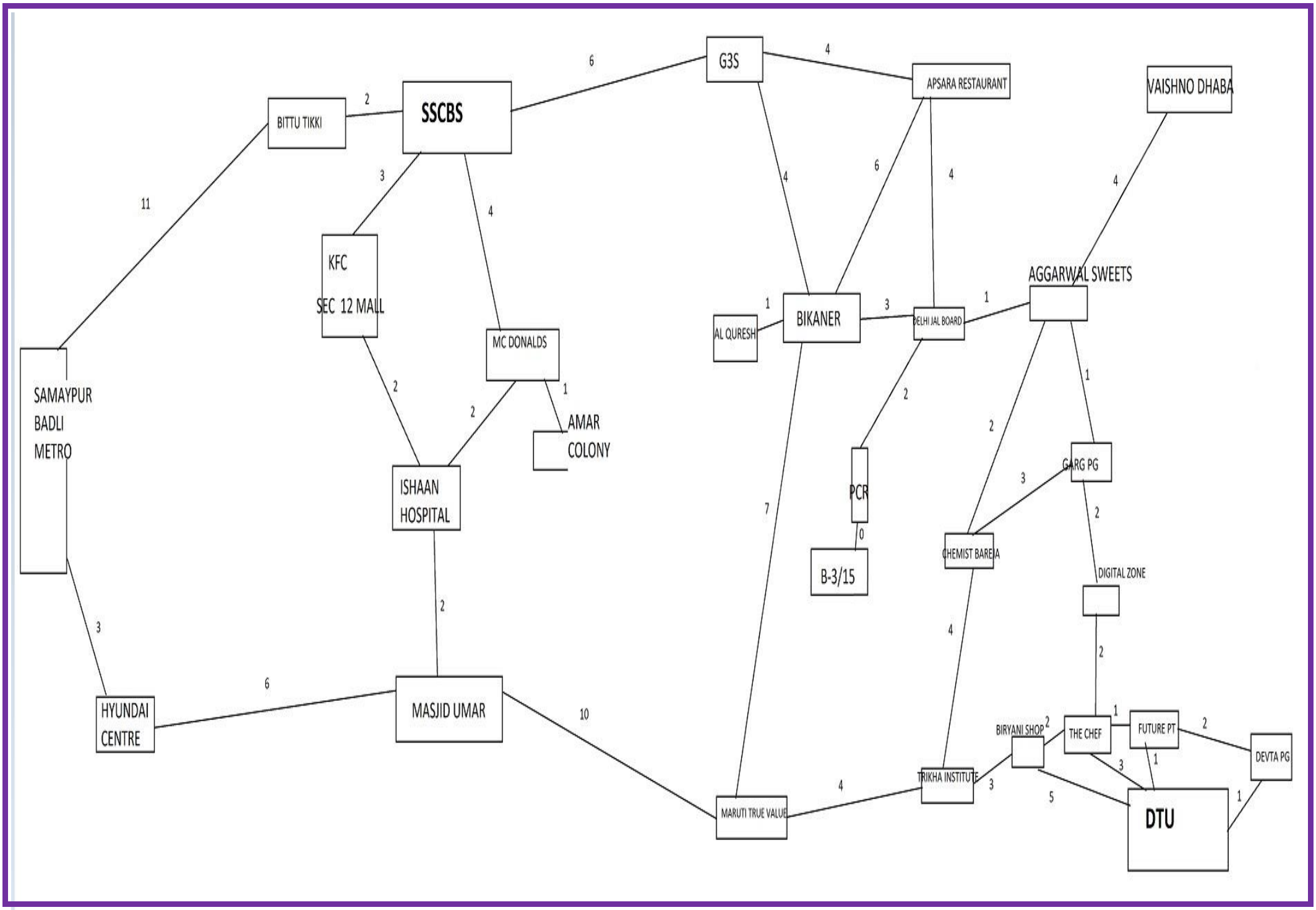
```
        Filter according to:
1. Price
2. ETA from your place
2
        Sorted from Lowest to Highest ETA

1  . THE_CHEF                                   Rs  350   2   mins
2  . BIRYANI_SHOP                               Rs  300   4   mins
3  . AGGARWAL_SWEETS                            Rs  200   7   mins
4  . VAISHO_DHABA                               Rs  250   11  mins
5  . BIKANER                                    Rs  450   11  mins
6  . APSARA_RESTAURANT                          Rs  400   12  mins
7  . AL_QURESHI                                 Rs  500   12  mins
8  . BITTU_TIKKI                                Rs  230   23  mins
9  . KFC_SEC12_MALL                             Rs  280   24  mins
10 . MC_DONALDS                                 Rs  220   25  mins
```

```
            Menu for AL_QURESHI
      1  . Afghani_Chicken                          155
      2  . Boneless_Butter_Chicken                  250
      3  . Boneless_Tawa_Chicken                    250
      4  . Butter_Chicken                           220
      5  . Butter_Naan                              40
      6  . Chicken_Fry                              115
      7  . Chicken_Kebab_Roll                       130
      8  . Chicken_Lollipop                         130
      9  . Chicken_Seekh_Kebab(6 Pcs)               155
      10 . Chicken_Tikka(6 Pcs)                     215
      11 . Chicken_Tikka_Roll                       150
      12 . Malai_Tikka(6 Pcs)                       220
      13 . Paneer_Tikka                             155
      14 . Plain_Naan                               25
      15 . Rumali_Roti                              10
      16 . Tandoori_Chicken                         140
      17 . Tandoori_Roti                            15
      18 . Tawa_Chicken                             220
```

```
      Enter the number of items you want to add
             5
      Enter serial number and quantity
      4 2
      Enter serial number and quantity
      18 1
      Enter serial number and quantity
      14 2
      Enter serial number and quantity
      14 1
      Enter serial number and quantity
      5 1
```

```
Your final Menu looks like:

      1  . Afghani_Chicken                          440
      2  . Boneless_Butter_Chicken                  220
      3  . Boneless_Tawa_Chicken                    50
      4  . Butter_Chicken                           25
      5  . Butter_Naan                              40

         Final Price to pay:                        775
```

```
      Do you have a discount coupon
      SUMEDHA_MAM_BEST
                    ***CONGRATS 25%  discount applied***
      Your final price is:                              581.25

             How would you like to make the payment

      1. Cash
      2. Credit/Debit Card
      3. Paytm / PhonePay / GooglePay
      2
      Enter card number and CVV
      746274828472 385
      Food is Preparing

      Our service will be at your doorstep in Expected Time(ETA): 12 minutes
      Our agent is coming via:
AL_QURESHI -> BIKANER -> DELHI_JAL_BOARD -> AGGARWAL_SWEETS -> GARG_PG -> DIGITAL_ZONE -> THE_CHEF -> DTU
PS C:\Users\kunj5\Desktop\programs>
```


SAMAYPUR
BADLI
METRO
BITTU TIKKI
SSCBS
G3S
APSARA RESTAURANT
VAISHNO DHABA
KFC
SEC 12 MALL
MC DONALDS
AMAR
COLONY
AL QURESH
BIKANER
DELHI JAL BOARD
AGGARWAL SWEETS
ISHAAN
HOSPITAL
PCR
B-3/15
GARG PG
CHEMIST BAREJA
DIGITAL ZONE
HYUNDAI
CENTRE
MASJID UMAR
MARUTI TRUE VALUE
TRIKHA INSTITUTE
BIRYANI SHOP
THE CHEF
FUTURE PT
DEVTA PG
DTU
11
6
2
3
4
4
6
4
4
1
3
1
2
2
1
2
7
0
2
3
2
4
6
2
10
4
3
2
1
2
3
1
5
1

CONCLUSIONS AND FUTURE WORK

- I successfully made a working food delivery project.
- For future planning, I would like to expand our project throughout the city and even beyond that.
- I have included several street restaurants and stalls, though their number is very high, I would like to expand it throughout.
- I learnt several new SSSP algorithms other than Dijkstra, which including bellman ford, learnt about multi-source Dijkstra, and how to print path using it.

REFERENCES

- https://cp-algorithms.com/
- https://codingblocks.com/
- https://www.geeksforgeeks.org/