C, C++, DSA in depth

Dijkstra's algorithm

Saurabh Shukla (MySirG)

Dijkstra's Algorithm

Dijkstra's Algorithm solves the single source shortest paths problem on a weighted directed graph $G = (V, E)$ for the case in which all edge weights are non negative

Thus $w(u, v) \geq 0$ for each edge $(u, v) \in E$

Dijkstra's algorithm maintain's a set S of vertices whose final shortest path weights from the source s have already been determined

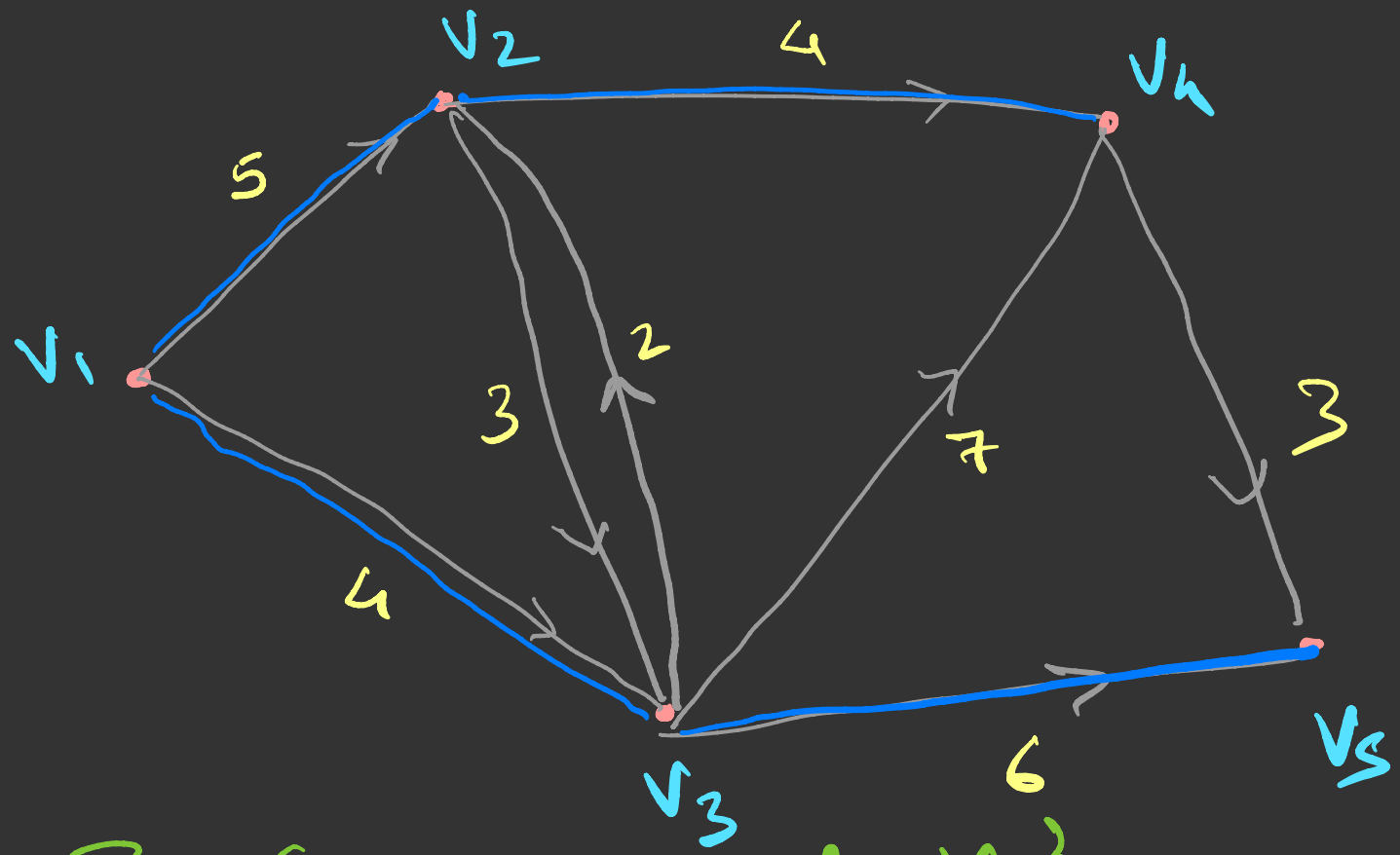
The algorithm repeatedly selects the vertex $u \in V - S$ with the minimum shortest path estimate, adds u to S, and relaxes all edges leaving u.

We uses a min priority queue Q of vertices, keyed by their d values.

Initially d value of each node is ∞, except the starting node, which has d value 0.

PAR maintains parent of particular node

Till the nodes are in Q their d value is ∞



	d	PAR
✗ v_1	0	–
✗ v_2	5	v_1
✗ v_3	4	v_1
✗ v_4	9	v_2
✗ v_5	10	v_3

$S = \{v_1, v_3, v_2, v_4, v_5\}$

if $(d[v] > d[u] + w(u,v))$

$v_1 \to v_2$ 5

$v_1 \to v_3$ 4

$v_1 \to v_2 \to v_4$ 9

$v_1 \to v_3 \to v_5$ 10

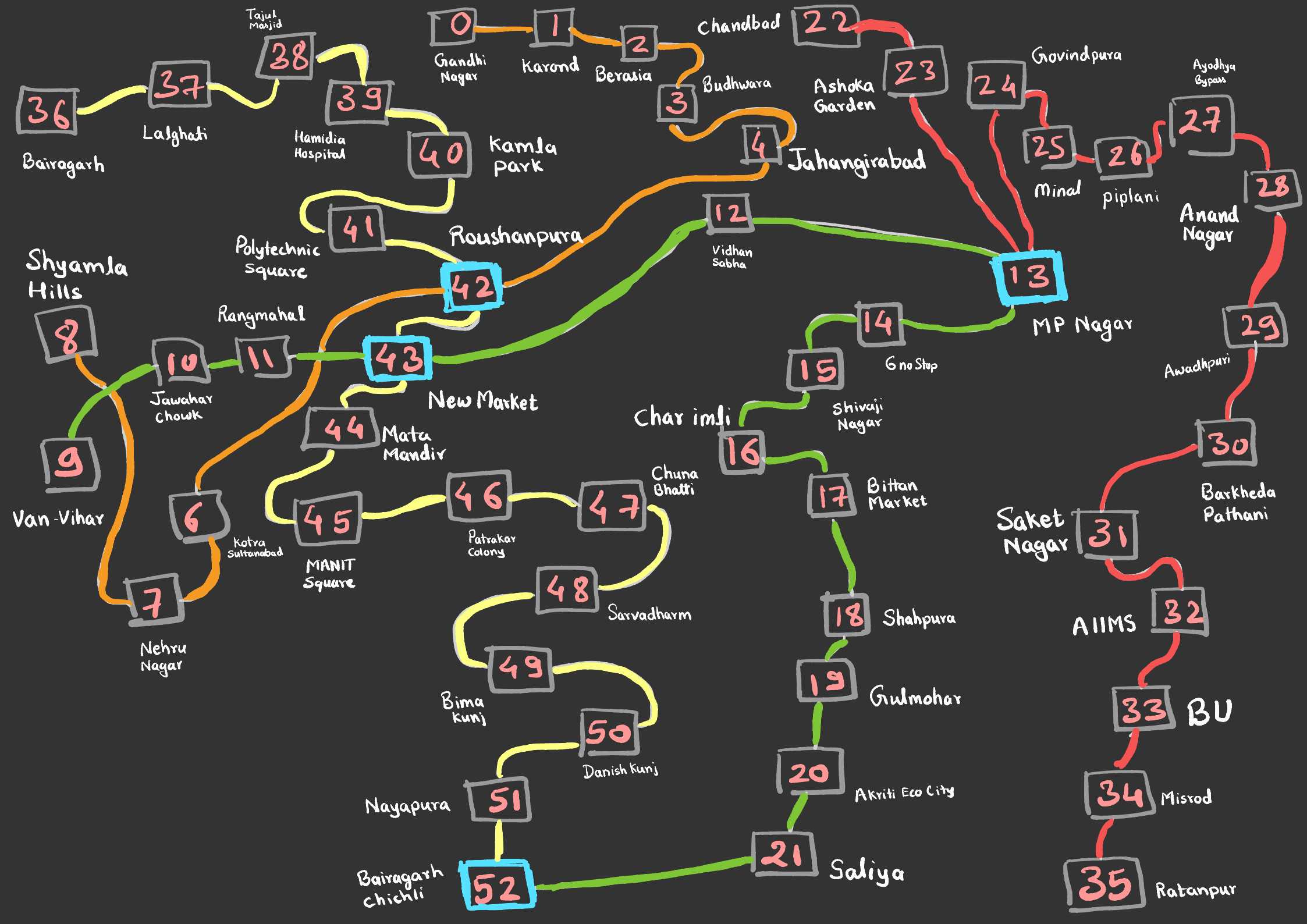
Relax(u, v, w)

1. if $d[v] > d[u] + w(u,v)$
   then $d[v] = d[u] + w(u,v)$
   $PAR[v] = u$
2. return

DIJKSTRA's (G, w, s)

1. initialize single source
2. $S \neq \emptyset$, $Q = V[G]$
3. Repeat steps 4 to 6 while $Q \neq \emptyset$
4. $u =$ extract-min(Q) , $S = S \cup \{u\}$
5. Repeat for each vertex $v \in Adj[u]$
6. Relax(u, v, w)
7. exit.

```
void dijkstra (int graph[53][53], int source, string stations[53])
{
    int distances[53];
    bool stat[53];
    for (int K=0; K<53; K++)
    {
        distance[K]= INT_MAX;
        stat[K] = false;
    }
    distance[source]=0;
    for (int K=0; K<53; K++)
    {
        int m = mindistance(distance, stat);
        stat[m]= true;
```


0
Gandhi Nagar
1
Karond
2
Berasia
Chandbad
22
3
Budhwara
Ashoka Garden
23
24
Govindpura
Ayodhya Bypass
27
25
Minal
26
piplani
28
Anand Nagar
Tajul Masjid
38
36
Bairagarh
37
Lalghati
39
Hamidia Hospital
40
Kamla park
4
Jahangirabad
41
Polytechnic Square
Roushanpura
42
12
Vidhan Sabha
13
MP Nagar
Shyamla Hills
8
Rangmahal
10
Jawahar Chowk
11
43
New Market
14
G no Stop
15
Shivaji Nagar
Char imli
16
17
Bittan Market
29
Awadhpuri
30
Barkheda Pathani
9
Van-Vihar
6
Kotra Sultanabad
44
Mata Mandir
45
MANIT Square
46
Patrakar Colony
47
Chuna Bhatti
48
Sarvadharm
7
Nehru Nagar
18
Shahpura
19
Gulmohar
20
Akriti Eco City
21
Saliya
Saket Nagar
31
AIIMS
32
33
BU
34
Misrod
35
Ratanpur
49
Bima Kunj
50
Danish Kunj
Nayapura
51
Bairagarh chichli
52

Station Codes

Station Code	Station Name
0	Gandhi Nagar
1	karond
2	Berasia
3	Budhwara
4	Jahangirabad
5	Roushanpura
6	kotra Sultanabad
7	Nehru Nagar
8.	Shyamla Hills
9	Van Vihar
10	Jawahar Chowk
11.	Rangmahal
12.	Vidhan Sabha
13.	MP Nagar
14.	6 no. stop
15.	Shivaji Nagar
16.	Char imli
17.	Bittan Market
18.	Shahpura
19.	Gulmohar
20.	Akriti Eco City
21.	Saliya
22.	Chandbad
23.	Ashoka Garden
24.	Govindpura
25.	Minal
26	Piplani

Station Code	Station Name
27.	Ayodhya Bypass
28.	Anand Nagar
29.	Awadhpuri
30.	Barkheda Pathani
31.	Saket Nagar
32.	AIIMS
33.	Barkatullah University
34.	Misrod
35.	Ratanpur
36.	Bairagarh
37.	Lalghati
38.	Tajul Masjid
39.	Hamidia Hospital
40.	Kamla Park
41.	Polytechnic Square
42.	Roshanpura
43.	New Market
44.	Mata Mandir
45.	MANIT Square
46.	Patrakar Colony
47.	Chuna Bhatti
48.	Sarvdharm
49.	Bima Kunj
50.	Danish Kunj
51.	Nayapura
52.	Bairagarh Chichli

Problem Statement

There are 53 Metro stations in BHOPAL Considering each station an integer value starting from 0 is assigned.

When the user enters the source station, as a result, user can see the list of all possible destination stations with the minimum number of stations between them.

You have to store information about the stations and their connections as given in the diagram in appropriate data structure.