

# **Jaypee Institute of Information Technology**



**TITLE:**

**SARS Cov-2 Vaccination System**

## **Details of Team**

- ❖ Anurag Singh 19103150
- ❖ Divyansh Bhargava 19103127

(Both from B-4 Batch)

## **Abstract of the project**

The project “SARS Cov-2 Vaccination System” is Computer Science model made using the applications of Data Structure and Algorithms.

As the name of this project suggests this model works for an objective to ease the vaccination drive for common people. In this system one can book a vaccination slot easily by uploading his/her details. The user will be asked to provide certain details regarding his/her identity and regarding the phase of vaccine he/she is applying for once the user is registered successfully, the system will display all the information regarding the location of the vaccination centres present nearby user and the user can book his/her slot for the vaccine depending on the availability of the vaccine on the particular centre which will also be displayed to the user at the time of booking.

Once the done with the selection of vaccination centre process the user will be registered for an automatic first come first jabbed vaccination process, the details of the same can be viewed in the waiting list, where user can remove the application if required due to some error in the application submitted and then that available slot will automatically be specified to the very next fellow present in the list.

Based on the phase and urgent need of some special case of user the preference can be given to him/her. It is possible that people who are applying for the vaccine under the foreign travel section might get the slot allotted earlier.

## Tools and Technologies used

- C++ Language
- Vectors
- Queue
- Sorting
- Graphs

## DATASET(KAGGLE) and few test cases run from our side

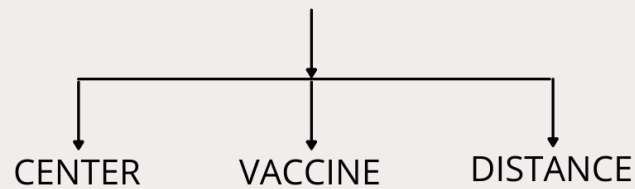
PATIENT ID	PATIENT NAME	PATIENT CONTACT NO.	Patient Phase
111	Anurag	12121	1
222	Divyansh	13131	2
333	Ram	14141	3
444	Shyam	15151	1
555	Karan	12123	3
666	Arjun	99999	2
777	Sita	16116	1
888	Geeta	13112	2
999	Piyush	11232	3
1110	Devarshi	12123	1
1221	Kelly, Mr. James	330911	1
1332	Wilkes, Mrs. James (Ellen Needs)	363272	2
1443	Myles, Mr. Thomas Francis	240276	2
1554	Wirz, Mr. Albert	315154	2
1665	Hirvonen, Mrs. Alexander (Helga E Lindqvist)	3101298	1
1776	Svensson, Mr. Johan Cervin	7538	3
1887	Connolly, Miss. Kate	330972	3
1998	Caldwell, Mr. Albert Francis	248738	1
2109	Abraham, Mrs. Joseph (Sophie Halaut Easu)	2657	3

2220	Davies, Mr. John Samuel	48871	1
2331	Ilieff, Mr. Ylio	349220	2
2442	Jones, Mr. Charles Cresson	694	2
	Snyder, Mrs. John Pillsbury (Nelle		
2553	Stevenson)	21228	1
2664	Howard, Mr. Benjamin	24065	3
	Chaffee, Mrs. Herbert Fuller (Carrie		
2775	Constance Toogood)	5734	2
	del Carlo, Mrs. Sebastiano (Argenia		
2886	Genovesi)	2167	1
2997	Keane, Mr. Daniel	233734	2
3108	Assaf, Mr. Gerios	2692	3
3219	Ilmakangas, Miss. Ida Livija	3101270	1
3330	Assaf Khalil, Mrs. Mariana (Miriam)"	2696	1
3441	Rothschild, Mr. Martin	17603	2
3552	Olsen, Master. Artur Karl	17368	2
3663	Flegenheim, Mrs. Alfred (Antoinette)	17598	2
3774	Williams, Mr. Richard Norris II	17597	3
	Ryerson, Mrs. Arthur Larned (Emily Maria		
3885	Borie)	17608	1
3996	Robins, Mr. Alexander A	53337	3
4107	Ostby, Miss. Helene Ragnhild	113509	1
4218	Daher, Mr. Shedid	2698	2
4329	Brady, Mr. John Bertram	113054	2
4440	Samaan, Mr. Elias	2662	1
4551	Louch, Mr. Charles Alexander	3085	3
4662	Jefferys, Mr. Clifford Thomas	31029	2
4773	Dean, Mrs. Bertram (Eva Georgetta Light)	2315	1
	Johnston, Mrs. Andrew G (Elizabeth Lily"		
4884	Watson)"	6607	2
4995	Mock, Mr. Philipp Edmund	13236	3
	Katavelas, Mr. Vassilios (Catavelas		
5106	Vassilios)"	2682	2
5217	Roth, Miss. Sarah A	342712	2
5328	Cacic, Miss. Manda	315087	1
5439	Sap, Mr. Julius	345768	3

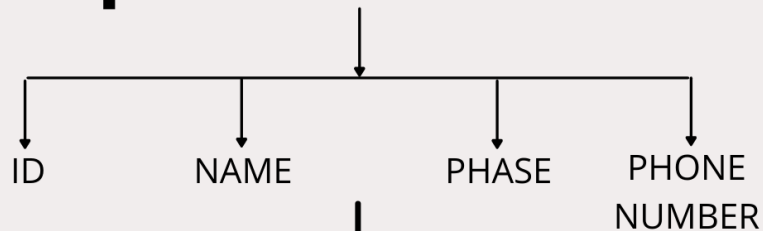
# **PROJECT DESIGN**

**Flowchart:**

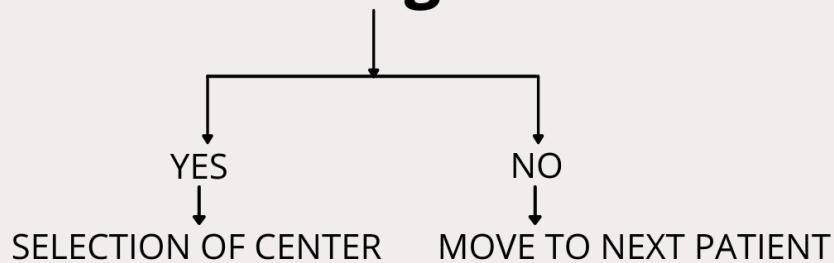
## **Input of Administerial Data**



## **Input of Patient Data**



## **Confirming Details**

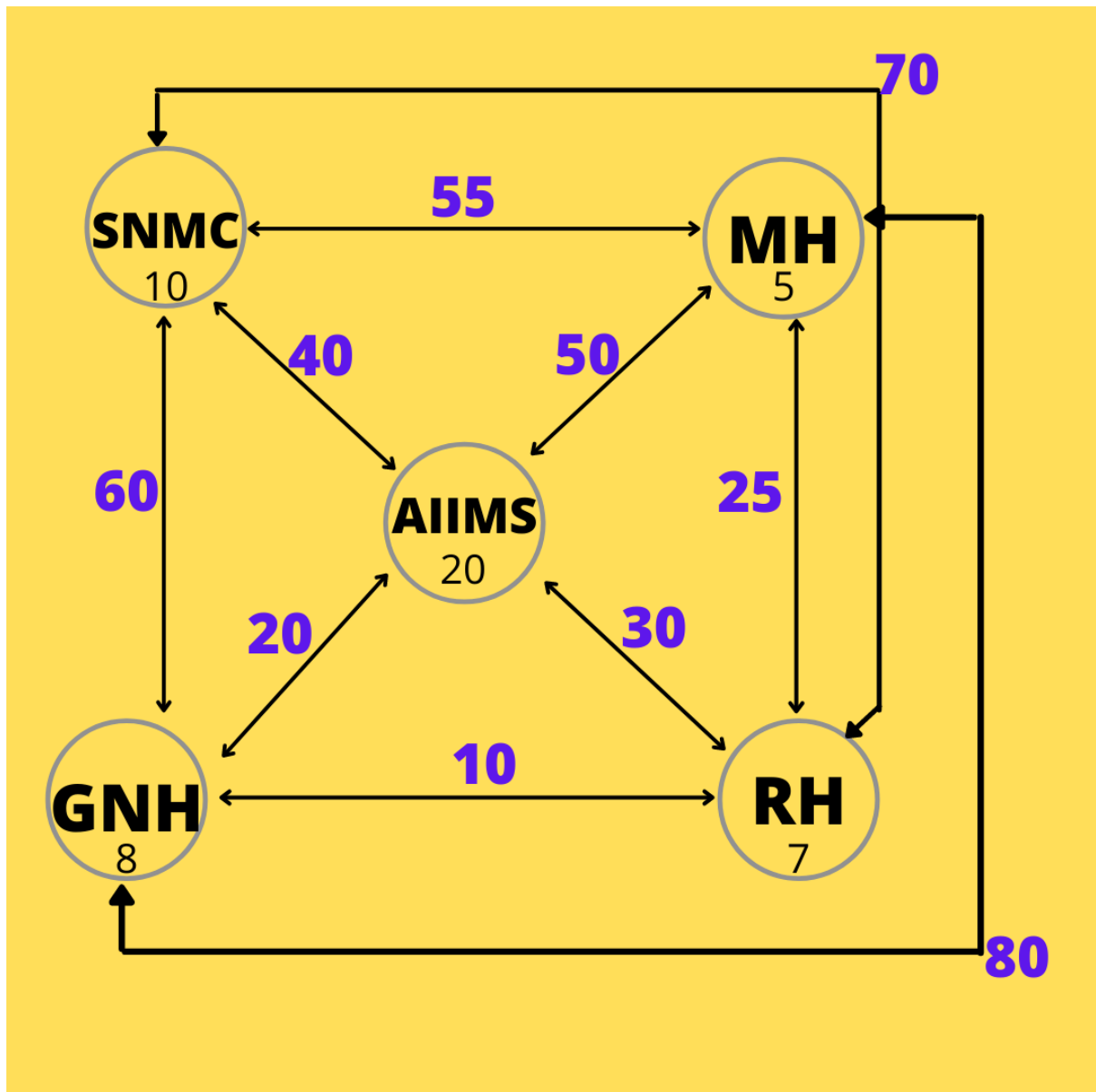


## **Confirming Center**

## **Slot Booked**

## **Printing Details**

Example Input Graph:



OUTPUT:



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
Enter the number of centers:
5
Enter the centers:
AIIMS
SNMC
RH
GNH
MH
Enter the number of vaccines:
AIIMS -> 20
SNMC -> 10
RH -> 7
GNH -> 8
MH -> 5
Enter the Graph
Weight/Distance of: AIIMS -> SNMC
40
Weight/Distance of: AIIMS -> RH
30
Weight/Distance of: AIIMS -> GNH
20
Weight/Distance of: AIIMS -> MH
50
Weight/Distance of: SNMC -> RH
70
Weight/Distance of: SNMC -> GNH
60
Weight/Distance of: SNMC -> MH
55
Weight/Distance of: RH -> GNH
10
Weight/Distance of: RH -> MH
25
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
Weight/Distance of: GNH -> MH
80
-----Enter Number Of Patients-----
5
ENTER THE DETAILS OF THE PATIENT
  ENTER Patient ID: 111
    Enter Patient name: Anurag
    Enter Patient Contact number: 12121
    Enter Patient Phase(1,2,3): 1
ENTER THE DETAILS OF THE PATIENT
  ENTER Patient ID: 222
    Enter Patient name: Divyansh
    Enter Patient Contact number: 13131
    Enter Patient Phase(1,2,3): 2
ENTER THE DETAILS OF THE PATIENT
  ENTER Patient ID: 333
    Enter Patient name: Ram
    Enter Patient Contact number: 14141
    Enter Patient Phase(1,2,3): 3
ENTER THE DETAILS OF THE PATIENT
  ENTER Patient ID: 444
    Enter Patient name: Shyam
    Enter Patient Contact number: 15151
    Enter Patient Phase(1,2,3): 1
ENTER THE DETAILS OF THE PATIENT
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
ENTER Patient ID: 555

Enter Patient name: Karan

Enter Patient Contact number: 123123

Enter Patient Phase(1,2,3): 3

---- OUTPUT----
The Patient(s) are in PHASE 3 :      333      555

The Patient(s) are in PHASE 1 :      111      444

The Patient(s) are in PHASE 2 :      222

---- OUTPUT----

The Patient(s)_id(s) ASSIGNED by PRIORITY PHASES(3,1,2) are :   333      555111      444      222

Patient ID: 333
Patient Name: Ram
Patient Contact Number: 14141
If you accept your slot enter Y otherwise N: Y

Enter the center-->  AIIMS

----OUTPUT----
The nearest vaccination centre from :AIIMS: to center :SNMC: is 40
The path is: SNMC-(10 Vaccines) <-AIIMS-(20 Vaccines)
Total Vaccines-30

----OUTPUT----
The nearest vaccination centre from :AIIMS: to center :RH: is 30
The path is: RH-(7 Vaccines) <-AIIMS-(20 Vaccines)
Total Vaccines-27
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
----OUTPUT----
The nearest vaccination centre from :AIIMS: to center :GNH: is 20
The path is: GNH-(8 Vaccines) <-AIIMS-(20 Vaccines)
Total Vaccines-28

----OUTPUT----
The nearest vaccination centre from :AIIMS: to center :MH: is 50
The path is: MH-(5 Vaccines) <-AIIMS-(20 Vaccines)
Total Vaccines-25

Choose center from above based on the number of vaccines available: AIIMS

SLOT BOOKED

ID: 333
Patient Name: Ram
Patient Chosen Center: AIIMS
Patient Phase: 3
Patient Contact Number: 14141
Security Code: 41

-----

Patient ID: 555
Patient Name: Karan
Patient Contact Number: 123123
If you accept your slot enter Y otherwise N: Y

Enter the center-->  SNMC

----OUTPUT----
The nearest vaccination centre from :SNMC: to center :AIIMS: is 40
The path is: AIIMS-(19 Vaccines) <-SNMC-(10 Vaccines)
Total Vaccines-29
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
----OUTPUT----
The nearest vaccination centre from :SNMC: to center :RH: is 70
The path is: RH-(7 Vaccines) <-SNMC-(10 Vaccines)
Total Vaccines-17

----OUTPUT----
The nearest vaccination centre from :SNMC: to center :GNH: is 60
The path is: GNH-(8 Vaccines) <-SNMC-(10 Vaccines)
Total Vaccines-18

----OUTPUT----
The nearest vaccination centre from :SNMC: to center :MH: is 55
The path is: MH-(5 Vaccines) <-SNMC-(10 Vaccines)
Total Vaccines-15

Choose center from above based on the number of vaccines available: GNH

SLOT BOOKED

ID: 555
Patient Name: Karan
Patient Chosen Center: GNH
Patient Phase: 3
Patient Contact Number: 123123
Security Code: 8467

Patient ID: 111
Patient Name: Anurag
Patient Contact Number: 12121
If you accept your slot enter Y otherwise N: N

Patient ID: 444
Patient Name: Shyam
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

Patient ID: 111
Patient Name: Anurag
Patient Contact Number: 12121
If you accept your slot enter Y otherwise N: N

Patient ID: 444
Patient Name: Shyam
Patient Contact Number: 15151
If you accept your slot enter Y otherwise N: Y

Enter the center--> MH

----OUTPUT----
The nearest vaccination centre from :MH: to center :AIIMS: is 50
The path is: AIIMS-(19 Vaccines) <-MH-(5 Vaccines)
Total Vaccines-24

----OUTPUT----
The nearest vaccination centre from :MH: to center :SNMC: is 55
The path is: SNMC-(10 Vaccines) <-MH-(5 Vaccines)
Total Vaccines-15

----OUTPUT----
The nearest vaccination centre from :MH: to center :RH: is 25
The path is: RH-(7 Vaccines) <-MH-(5 Vaccines)
Total Vaccines-12

----OUTPUT----
The nearest vaccination centre from :MH: to center :GNH: is 35
The path is: GNH-(7 Vaccines) <-RH-(7 Vaccines) <-MH-(5 Vaccines)
Total Vaccines-19
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
Total Vaccines-19

Choose center from above based on the number of vaccines available: RH

SLOT BOOKED

ID: 444
Patient Name: Shyam
Patient Chosen Center: RH
Patient Phase: 1
Patient Contact Number: 15151
Security Code: 6334

Patient ID: 222
Patient Name: Divyansh
Patient Contact Number: 13131
If you accept your slot enter Y otherwise N: Y

Enter the center--> GNH

---OUTPUT---
The nearest vaccination centre from :GNH: to center :AIIMS: is 20
The path is: AIIMS-(19 Vaccines) <-GNH-(7 Vaccines)
Total Vaccines-26

---OUTPUT---
The nearest vaccination centre from :GNH: to center :SNMC: is 60
The path is: SNMC-(10 Vaccines) <-GNH-(7 Vaccines)
Total Vaccines-17

---OUTPUT---
The nearest vaccination centre from :GNH: to center :RH: is 10
The path is: RH-(6 Vaccines) <-GNH-(7 Vaccines)
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

Enter the center--> GNH

---OUTPUT---
The nearest vaccination centre from :GNH: to center :AIIMS: is 20
The path is: AIIMS-(19 Vaccines) <-GNH-(7 Vaccines)
Total Vaccines-26

---OUTPUT---
The nearest vaccination centre from :GNH: to center :SNMC: is 60
The path is: SNMC-(10 Vaccines) <-GNH-(7 Vaccines)
Total Vaccines-17

---OUTPUT---
The nearest vaccination centre from :GNH: to center :RH: is 10
The path is: RH-(6 Vaccines) <-GNH-(7 Vaccines)
Total Vaccines-13

---OUTPUT---
The nearest vaccination centre from :GNH: to center :MH: is 35
The path is: MH-(5 Vaccines) <-RH-(6 Vaccines) <-GNH-(7 Vaccines)
Total Vaccines-18

Choose center from above based on the number of vaccines available: MH

SLOT BOOKED

ID: 222
Patient Name: Divyansh
Patient Chosen Center: MH
Patient Phase: 2
Patient Contact Number: 13131
Security Code: 6500
```

# IMPLEMENTATION DETAILS

## CODE:

```
#include <iostream>
#include<queue>
#include<vector>
#include<stack>
#include<string>
#include<cmath>
#include<map>
#include<bits/stdc++.h>
```

```

using namespace std;

class Patient{
public:
    int patient_id;
    string p_name;
    int p_contact;
    int P_condition;

    Patient(){
        cout<<"ENTER THE DETAILS OF THE PATITENT"<<endl;
        cout<<"      ENTER Patient ID: ";
        cin>>patient_id;
        cout<<"\n      Enter Patient name: ";
        cin>>p_name;
        cout<<"\n      Enter Patient Contact number: ";
        cin>>p_contact;
        //1->critical
        //2->below critical
        //3->normal
        cout<<"\n      Enter Patient Phase(1,2,3): ";
        cin>>P_condition;
    }
    //GET PATIENT DETAILS->

    int getPatientCondition()
    {
        return P_condition;
    }

    int getPatientID(){
        return patient_id;
    }

    int getPatientContact(){
        return p_contact;
    }

    string getPatientName(){
        return p_name;
    }
    void getPatientDetails()
    {
        cout<<"\n\tPatient ID: "<<patient_id;
        cout<<"\n\tPatient Name: "<<p_name;
        cout<<"\n\tPatient Contact Number: "<<p_contact;
    }
}

```

```

};

// void getPatientDetailsWithPatientID(int p_id,n,)
// {
//     for(int i=0;i<n;i++)
//     {
//         if(P[i]->patient_id==p_id)
//         {
//             P[i]->getPatientDetails();
//         }
//     }
// }

#define MAX 10
#define INF 100000;

void dijk(int G[MAX][MAX], int n, int start, vector<string> centers, vector<int> Vaccine)
{
    int cost[MAX][MAX], dist[MAX], visited[MAX], pred[MAX];
    int i, j, count, mindist, nextnode;

    for (i = 0; i < n; i++)
    {
        for (j = 0; j < n; j++)
        {
            if (G[i][j] == 0)
            {
                cost[i][j] = INF;
            }
            else
            {
                cost[i][j] = G[i][j];
            }
        }
    }
    for (i = 0; i < n; i++)
    {
        dist[i] = cost[start][i];
        pred[i] = start;
        visited[i] = 0;
    }
    dist[start] = 0;
    visited[start] = 1;
    count = 1;

```

```

while (count < n - 1)
{
    mindist = INF;
    for (i = 0; i < n; i++)
    {
        if (dist[i] < mindist && !visited[i])
        {
            mindist = dist[i];
            nextnode = i;
        }
    }
    visited[nextnode] = 1;

    for (i = 0; i < n; i++)
    {
        if (!visited[i])
        {
            if ((mindist + cost[nextnode][i]) < dist[i])
            {
                dist[i] = mindist + cost[nextnode][i];
                pred[i] = nextnode;
            }
        }
    }
    count++;
}

for (i = 0; i < n; i++)
{
    int sum = 0;
    if (i != start)
    {
        cout << "\n ----OUTPUT----\n";
        cout << "The nearest vaccination centre from :" << centers[start]
<< ": to center :" << centers[i] << " is " << dist[i] << endl;
        cout << "The path is: " << centers[i] << "-
(" << Vaccine[i] << " Vaccines) ";
        sum = sum + Vaccine[i];
        j = i;
        do
        {
            j = pred[j];
            cout << "<" << centers[j] << "-
(" << Vaccine[j] << " Vaccines) ";
            sum = sum + Vaccine[j];
        } while (j != start);
    }
    cout << "\n";
}

```

```

        if(sum!=0){
            cout << "Total Vaccines-" << sum << endl;
        }
    }
}

int getIndex(vector<string> centers, int n)
{
    string x;
    int ind;
    cout << "\n\tEnter the center--> ";
    cin >> x;

    for (int i = 0; i != n; i++)
    {
        if (centers[i] == x)
        {
            ind = i;
            break;
        }
    }
    return ind;
}

void getGraph(int G[MAX][MAX], vector<string> centers, int n)
{
    for (int i = 0; i < n; i++)
    {
        for (int j = i; j < n; j++)
        {
            if (i != j)
            {
                int X;
                cout << "Weight/Distance of: " << centers[i] << " -
> " << centers[j] << endl;
                cin >> X;
                G[i][j] = G[j][i] = X;
            }
            else
            {
                G[i][j] = 0;
            }
        }
    }
}

void showq(queue<int> gq)

```



```

{
    queue<int> g = gq;
    while (!g.empty()) {
        cout << '\t' << g.front();
        g.pop();
    }
    cout << '\n';
}

void makePatientQueue(queue<int> q1,queue<int> q2,queue<int> q3,queue<int> patientQueue)
{
    queue<int> g1 = q1;
    queue<int> g2 = q2;
    queue<int> g3 = q3;

    while (!g1.empty()) {
        patientQueue.push(g1.front());
        g1.pop();
    }
    while (!g2.empty()) {
        patientQueue.push(g2.front());
        g2.pop();
    }
    while (!g3.empty()) {
        patientQueue.push(g3.front());
        g3.pop();
    }
}

// string chooseCenter(vector<string> center,vector<int> vaccine,int n)
// {

//     return cc;
// }

void displaydetails(string cc,int pid)
{

}

int main()
{
    system("CLS");
    //VACCINES centers
    int G[MAX][MAX], n, start;

```

```

int i, j;

cout << "\n";
cout << "Enter the number of centers:" << endl;
cin >> n;
cout << "\n";

//centers name
cout << "Enter the centers:" << endl;
vector<string> center(10);
for (int i = 0; i < n; i++)
{
    cin >> center[i];
}
cout << "\n";
cout << "Enter the number of vaccines:" << endl;
vector<int> vaccine(10);
for (int i = 0; i < n; i++)
{
    cout << center[i] << " -> ";
    cin >> vaccine[i];
}

cout << "\n";
cout << "Enter the Graph" << endl;
getGraph(G, center, n);


int m;
cout<<" -----Enter Number Of Patients-----"<<endl;
cin>>m;
// patients details
Patient *P[m];

for(int i=0;i<m;i++)
{
    P[i]=new Patient;
}

//# QUEUE A,B,C
queue<int> q1;
queue<int> q2;
queue<int> q3;
queue<int> patientQueue;
cout<<endl;
for(int i=0;i<m;i++){

```

```

        if(P[i]->P_condition==3)
        {
            // cout<<"\npatient id:"<<P[i]->patient_id;
            int p=P[i]->patient_id;
            q1.push(p);
        }
        else if(P[i]->P_condition==1){
            // cout<<"\npatient id:"<<P[i]->patient_id;
            int p=P[i]->patient_id;
            q2.push(p);
        }
        else if(P[i]->P_condition==2){
            // cout<<"\npatient id:"<<P[i]->patient_id;
            int p=P[i]->patient_id;
            q3.push(p);
        }
    }
}
cout<<"\n    ---- OUTPUT----";
cout << "\nThe Patient(s) are in PHASE 3 : ";
showq(q1);
cout << "\nThe Patient(s) are in PHASE 1 : ";
showq(q2);
cout << "\nThe Patient(s) are in PHASE 2 : ";
showq(q3);
cout<<"\n    ---- OUTPUT----\n";
//MAKING A QUEUE THAT STORES PATIENT IDS WITH CONDITION
// makePatientQueue(q1,q2,q3,patientQueue);
queue<int> g1 = q1;
queue<int> g2 = q2;
queue<int> g3 = q3;
while (!g1.empty()) {
    patientQueue.push(g1.front());
    g1.pop();
}
while (!g2.empty()) {
    patientQueue.push(g2.front());
    g2.pop();
}
while (!g3.empty()) {
    patientQueue.push(g3.front());
    g3.pop();
}

cout << "\nThe Patient(s)_id(s) ASSIGNED by PRIORITY PHASES(3,1,2) are : "
;

showq(patientQueue);
// getPatientDetailsWithPatientID(111,3);

```

```

queue<int> patientIDS=patientQueue;
while(!patientIDS.empty()){

    char choice;

    int PidQueue=patientIDS.front();
    for(int i=0;i<m;i++)
    {
        if(P[i]->patient_id==PidQueue)
        {
            P[i]->getPatientDetails();
            cout<<"\nIf you accept your slot enter Y otherwise N: ";
            cin>>choice;
            if(choice=='Y'){
                start=getIndex(center,n);

                dijk(G, n, start, center, vaccine);
                string cc;
                cout<<"\n    Choose center from above based on the number of va
ccines available: ";
                cin>>cc;
                for(int j=0;j<n;j++)
                {
                    if(cc==center[j])
                    {
                        vaccine[j]-=1;
                        // cout<<"\nCOMPARED ANURAG DIVYANSH-----
-----";

                        break;
                    }
                }
                cout<<"\n    SLOT BOOKED  \n";

// function call
                cout<<"\nID: "<<P[i]->getPatientID()<<endl;
                cout<<"Patient Name: "<<P[i]->getPatientName()<<endl;
                cout<<"Patient Chosen Center: "<<cc<<endl;
                cout<<"Patient Phase: "<<P[i]->getPatientCondition()<<endl;
                cout<<"Patient Contact Number: "<<P[i]-
>getPatientContact()<<endl;
                cout<<"Security Code: "<<rand()%10000<<endl;
                cout<<"\n_____ \n";
                patientIDS.pop();
            }
            else{
                patientIDS.pop();
            }
        }
    }
}

```

```
    }  
    }  
    cout<<endl;  
  
}  
  
return 0;  
}
```

## REFERENCES:

- **Let us C/CPP(Book)**
- **Geeks For Geeks**
- **Kaggle**
- **Google**
- **College Slides**
- **Tutorials point**
- **Beginning C++**
- **C++ STL Libraries**
- **YouTube**
- **Programmiz**

# Declaration Form:

Team Number	1
Project Title	SARS Cov-2 Vaccination System
Batch Number	B4
File Name	1_SARS_Cov-2_Vaccination_System_.dox

## Contribution Details

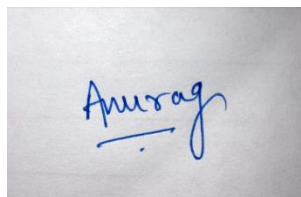
Sno	Roll Number	Name	E-mail	Contribution in this work (write your contribution in this work such as task done, tools explored, knowledge gained and presented, etc.)
1	19103150	Anurag Singh	<a href="mailto:19103150@mail.jiit.ac.in">19103150@mail.jiit.ac.in</a>	I, Anurag contributed in this project by presenting this project and learned many new things about data structures, graphs, Classes and Objects, algorithms, dynamic programming and got a chance to explore deep into the field of computer science.
2	19103127	Divyansh Bhargava	<a href="mailto:19103127@mail.jiit.ac.in">19103127@mail.jiit.ac.in</a>	I, Divyansh contributed in this project by presenting this project and learned many new things about OOPs, C++, DSA and specially graphs. I explored various applications of dijkstra while working on this project.

## Declaration

I/We hereby declare that this submission is my/our own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where

due acknowledgment has been made in the text. I/We accept the use of the material presented in this report for Education/Research/Teaching purpose by the faculty.

**Signature**



**Name : Anurag Singh**

**Name : Divyansh Bhargava**

**Date & Place: 20-July-2021 –WFM project  
– WFM Project**

**Date & Place: 20-July-2021**