



CHAT SURVEILLANCE PROGRAM



www.github.com/8nkit/DAAproject



DECRYPTING IT.

Submitted by:



MAVERICK'S



O I may be slow to respond.

Abhinav Joshi 8817103002

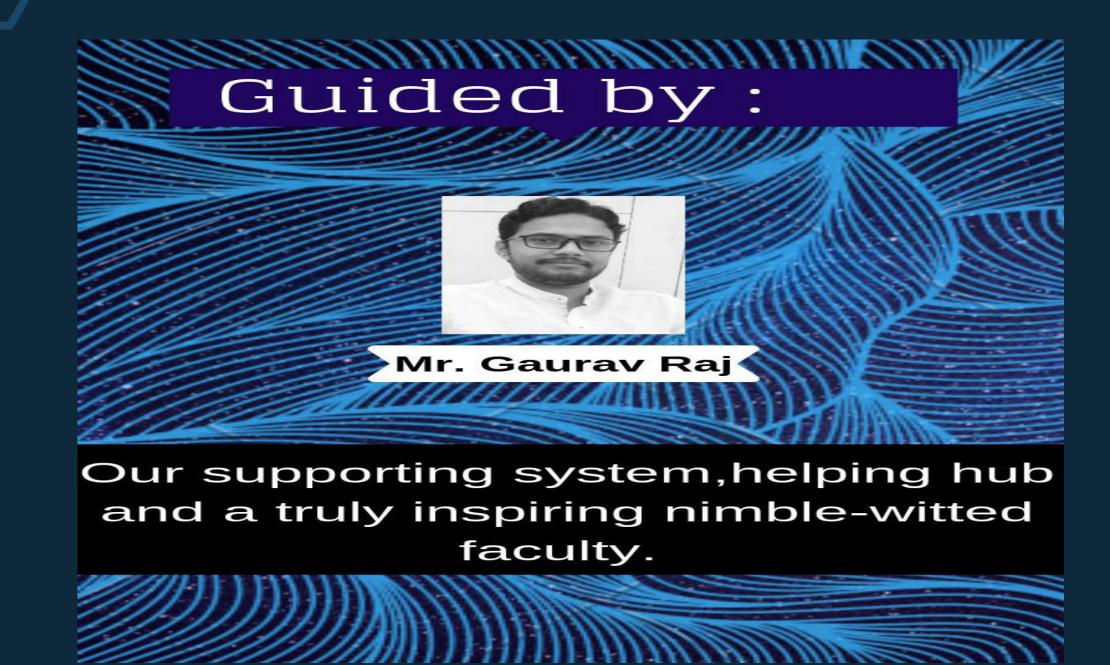


Sophomore

Ankit Sharma 8817103009



Piyush Kansal 8817103023



Sort arraylist of no's ***Descending order Load Fetching Searching Sorting Start Chat File Search for the 10 digit no.'s Get the top no. Working Store them in a arraylist * Element stored at first position Tracking in the arraylist Get path & distance Using Dijkstra's algo As GURGAON is assumed to be source location and Generate a file containing shortest path of the System and distances between all cities location obtained from tracker is destination Get Generate **OPTION 1** Fetching Location map Using the dummy tracker Storing **Generating random location the output Save the encrypted data Using modification of AES Phone number as well as the shortest distance and Encrypt the stored output by using path encrypted then saved to a txt file modification of AES by adding reversal

Encryption



Save it to

text file

E-mail

E-MAIL SENT



Using JAVA SMTP

Attach the text file to the mail and send it with subject MAVERICK'S







Used Algorithms

- 1. Searching
- 2. Sorting
- 3. Dijkstra's Algorithm
- 4. Min heap
- **5. JAVA AES**
- 6. Most occurred longest substring (DP)
- 7. Reversal of String (Divide & Conquer)



Code snippets

```
package finalproject;
import java.util.*;
public class Graph {
 private ArrayList<Vertex> vertices;
    public Graph(int numberVertices){
       vertices = new ArrayList<Vertex>(numberVertices);
       for(int i=0;i<numberVertices;i++){</pre>
            vertices.add(new Vertex(Integer.toString(i)));
    public void addEdge(int src, int dest, int weight){
       Vertex s = vertices.get(src);
       Edge new_edge = new Edge(vertices.get(dest), weight);
       s.neighbours.add(new edge);
    public ArrayList<Vertex> getVertices() {
       return vertices;
    public Vertex getVertex(int vert){
       return vertices.get(vert);
```

```
static void sendmail(){
      Scanner sc= new Scanner(System.in);
      System.out.println("\n ");
      System.out.println("Recipent email : ");
         String to = sc.nextLine();
  System.out.print("To send this mail you must login to your mailing account : \n");
      System.out.print("Email : ");
      String from = sc.nextLine();
      System.out.print("Password : *******");
      String pass="icing9cake@";
      System.out.println("\n");
       System.out.println("Logging you in and sending your mail ...");
       int timeToWait = 3; //second
           for (int i=0; i<timeToWait ; i++) {</pre>
               Thread.sleep(400);
               System.out.print(".");
         catch (InterruptedException ie)
           Thread.currentThread().interrupt();
```

```
package finalproject;
import java.util.*;
public class Vertex implements Comparable<Vertex>{
 public final String name;
    public ArrayList<Edge> neighbours;
    public LinkedList<Vertex> path;
    public double minDistance = Double.POSITIVE INFINITY;
    public Vertex previous;
    public int compareTo(Vertex other){
        return Double.compare(minDistance,other.minDistance);
    public Vertex(String name){
       this.name = name;
       neighbours = new ArrayList<Edge>();
       path = new LinkedList<Vertex>();
    public String toString(){
       return name;
```



Credits

These awesome resources helped us a lot while this project was in making :

- ♦ Introduction to Algorithms (By : Cormen)
- StackOverflow (Some high rated and working answers)





Thanks!

