

SKRIPSI

PERANGKAT LUNAK LOGIN OTOMATIS
UNTUK *CAPTIVE PORTAL* WI-FI



YOHANES MARIO CHANDRA

NPM: 2011730031

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI DAN SAINS
UNIVERSITAS KATOLIK PARAHYANGAN
2016

UNDERGRADUATE THESIS

**AUTOMATED LOGIN SOFTWARE
FOR WI-FI CAPTIVE PORTAL**



YOHANES MARIO CHANDRA

NPM: 2011730031

**DEPARTMENT OF INFORMATICS
FACULTY OF INFORMATION TECHNOLOGY AND SCIENCES
PARAHYANGAN CATHOLIC UNIVERSITY
2016**

LEMBAR PENGESAHAN

**PERANGKAT LUNAK LOGIN OTOMATIS
UNTUK *CAPTIVE PORTAL* WI-FI**

YOHANES MARIO CHANDRA

NPM: 2011730031

Bandung, 1 Agustus 2016

Menyetujui,

Pembimbing Tunggal

Pascal Alfadian, M.Comp.

Ketua Tim Penguji

Anggota Tim Penguji

«penguji 1»

«penguji 2»

Mengetahui,

Ketua Program Studi

Mariskha Tri Adithia, P.D.Eng

PERNYATAAN

Dengan ini saya yang bertandatangan di bawah ini menyatakan bahwa skripsi dengan judul:

PERANGKAT LUNAK LOGIN OTOMATIS UNTUK *CAPTIVE PORTAL* WI-FI

adalah benar-benar karya saya sendiri, dan saya tidak melakukan penjiplakan atau pengutipan dengan cara-cara yang tidak sesuai dengan etika keilmuan yang berlaku dalam masyarakat keilmuan.

Atas pernyataan ini, saya siap menanggung segala risiko dan sanksi yang dijatuhkan kepada saya, apabila di kemudian hari ditemukan adanya pelanggaran terhadap etika keilmuan dalam karya saya, atau jika ada tuntutan formal atau non-formal dari pihak lain berkaitan dengan keaslian karya saya ini.

Dinyatakan di Bandung,
Tanggal 1 Agustus 2016

Meterai

Yohanes Mario Chandra
NPM: 2011730031

ABSTRAK

«Tuliskan abstrak anda di sini, dalam bahasa Indonesia» Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetur.

Kata-kata kunci: «Tuliskan di sini kata-kata kunci yang anda gunakan, dalam bahasa Indonesia»

ABSTRACT

«Tuliskan abstrak anda di sini, dalam bahasa Inggris» Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetur.

Keywords: «Tuliskan di sini kata-kata kunci yang anda gunakan, dalam bahasa Inggris»

«kepada siapa anda mempersembahkan skripsi ini...?»

KATA PENGANTAR

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Bandung, Agustus 2016

Penulis

DAFTAR ISI

KATA PENGANTAR	xv
DAFTAR ISI	xvii
DAFTAR GAMBAR	xviii
DAFTAR TABEL	xix
1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	1
1.3 Tujuan Penelitian	2
1.4 Batasan Masalah	2
1.5 Metodologi Penelitian	2
1.6 Sistematika Pembahasan	3
2 THE DESCRIPTION OF A SET OF TRAJECTORIES AND ITS MEDIAN TRAJECTORY	5
2.1 Set of Trajectories	5
2.2 Properties of the Median Trajectory	6
DAFTAR REFERENSI	9
A THE PROGRAM	11
B THE SOURCE CODE	13

DAFTAR GAMBAR

2.1	Numbering of points and segments	5
2.2	Possible median trajectory (in black) with backward direction (pointed by the blue arrow)	6
2.3	Red arcs indicate the angular change at each vertex	6
A.1	Interface of the program	11

DAFTAR TABEL

2.1	Table Γ after inserting \mathcal{S}_1	7
2.2	\mathcal{S}_2 between v_{start} and \mathcal{S}_1	7
2.3	\mathcal{S}_2 between \mathcal{S}_1 and v_{end}	7
2.4	Final Γ	7

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Internet adalah salah satu hal yang sulit dipisahkan dari keseharian manusia masa kini. Salah satu cara seseorang dapat mengakses internet adalah dengan menggunakan teknologi Wi-Fi. Wi-Fi mengharuskan pengguna terhubung pada suatu *access-point*. *Access-point* tersebut dapat memiliki dua status, yaitu terproteksi atau tidak terproteksi. Proteksi pada *access-point* dapat dilakukan dengan beberapa cara, yaitu menggunakan protokol IEEE 802.11, atau menggunakan *captive portal*. Alat yang sudah pernah terhubung dengan *access-point* yang diproteksi dengan protokol IEEE 802.11 akan dengan mudah terhubung kembali dengan *access-point* tersebut karena alat tersebut biasanya sudah menyimpan password untuk *access-point* yang bersangkutan. Alat yang akan terhubung dengan *access-point* yang diproteksi menggunakan *captive portal* belum memiliki cara untuk mengingat username dan password untuk *captive portal* tersebut sehingga login otomatis belum dapat dilakukan untuk *access-point* jenis ini.

Berdasarkan pengamatan peneliti, *captive portal* banyak digunakan untuk proteksi *access point* pada tempat-tempat umum seperti lingkungan universitas, *starbucks*, *Mc Donald's*, dan beberapa tempat yang dapat diakses melalui *@wifi.id*, *free@wifi.id* dan *access point* sejenis. Oleh karena itu, dibutuhkan mekanisme yang bisa membantu proses login untuk *access-point* tipe ini. Terdapat dua cara untuk menciptakan mekanisme ini, yaitu dengan mengintegrasikannya dengan sistem operasi, atau menggunakan perangkat lunak pihak ketiga. Untuk dapat melakukan pengintegrasian mekanisme tersebut dengan sistem operasi, dibutuhkan akses kepada kode sumber sistem aplikasi tersebut. Oleh karena itu, pilihan yang lebih bijak sebagai seseorang yang tidak memiliki akses tersebut adalah dengan menciptakan perangkat lunak pihak ketiga.

1.2 Rumusan Masalah

Rumusan masalah yang akan dibahas pada penelitian ini adalah sebagai berikut:

- Bagaimana caranya melakukan implementasi login otomatis pada *captive portal* yang memiliki tingkat kenyamanan yang setara dengan login otomatis pada proteksi Wi-Fi berbasis protokol IEEE 802.11?
- Apa saja yang perlu dilakukan untuk mengamankan username dan password yang disimpan oleh user?

- Informasi apa saja yang dibutuhkan untuk menciptakan identitas unik untuk setiap *captive portal* pada jaringan yang berbeda?

1.3 Tujuan Penelitian

Tujuan penelitian ini adalah sebagai berikut:

- Melakukan implementasi login otomatis pada *captive portal* yang memiliki tingkat kenyamanan yang setara dengan login otomatis pada proteksi Wi-Fi berbasis protokol IEEE 802.11.
- Memastikan username dan password pengguna disimpan secara aman?
- Menentukan informasi yang dibutuhkan untuk menciptakan identitas unik untuk setiap *captive portal* pada jaringan yang berbeda?

1.4 Batasan Masalah

Batasan masalah dari penelitian ini adalah sebagai berikut:

- Perangkat lunak dibangun untuk sistem operasi Windows 8 sampai dengan Windows 10.
- Perangkat lunak dibangun menggunakan bahasa pemrograman C#.
- Elemen keamanan informasi yang akan diimplementasikan pada perangkat lunak ini adalah enkripsi username dan password yang disimpan oleh user.

1.5 Metodologi Penelitian

Metodologi penelitian yang akan dilakukan adalah sebagai berikut:

1. Melakukan studi literatur mengenai hal-hal yang berkaitan dengan perancangan dan pembuatan aplikasi, yaitu:
 - Pemrograman menggunakan bahasa pemrograman C#.
 - Penggunaan kelas `WebBrowser` pada C#.
 - Penggunaan objek `PasswordVault` pada C#.
 - Sistem login Wi-Fi berbasis web pada *access-point* dari berbagai merek.
2. Melakukan analisis perangkat lunak sejenis.
3. Melakukan analisis kebutuhan untuk mengimplementasikan mekanisme login otomatis ini.
4. Merancang perangkat lunak login otomatis ini.
5. Melakukan implementasi hasil rancangan dengan bahasa pemrograman C# pada sistem operasi Windows 10.

6. Melakukan pengujian terhadap perangkat lunak untuk menghasilkan perbaikan terhadap perangkat lunak tersebut.
7. Membuat kesimpulan dari hasil penelitian dan saran untuk penelitian selanjutnya.

1.6 Sistematika Pembahasan

Laporan skripsi ini terdiri dari beberapa bab, yaitu:

1. Bab Pendahuluan
Bab ini berisi latar belakang, rumusan masalah, tujuan penelitian, batasan masalah, metodologi penelitian, dan sistematika pembahasan.
2. Bab Dasar Teori
Bab ini berisi dasar-dasar pemrograman menggunakan bahasa pemrograman C#, dokumentasi kelas WebBrowser, dokumentasi objek PasswordVault, sistem login pada Mikrotik, dan sistem login pada Cisco.
3. Bab Analisis
Bab ini berisi analisis kebutuhan untuk perancangan dan pembuatan perangkat lunak login otomatis untuk proteksi Wi-Fi berbasis web.
4. Bab Perancangan
Bab ini berisi perancangan perangkat lunak login otomatis untuk proteksi Wi-Fi berbasis web.
5. Bab Implementasi dan Pengujian
Bab ini berisi implementasi perangkat lunak login otomatis untuk proteksi Wi-Fi berbasis web beserta pengujian dan hasil perbaikannya.
6. Bab Kesimpulan dan Saran
Bab ini berisi kesimpulan dari penelitian ini dan saran yang akan diberikan untuk penelitian selanjutnya.

BAB 2

THE DESCRIPTION OF A SET OF TRAJECTORIES AND ITS MEDIAN TRAJECTORY

2.1 Set of Trajectories

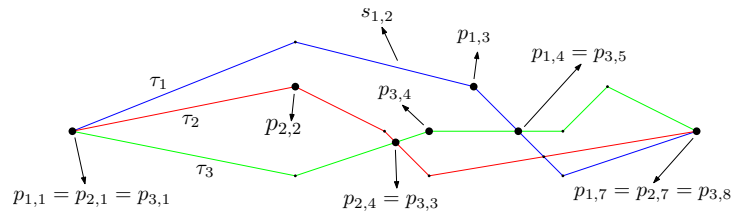
In this thesis, we only consider the spatial component of the trajectory. Therefore, we represent a trajectory as a polygonal line built by a series of points and connected by line segments.

Let $T := \{\tau_1, \tau_2, \dots, \tau_m\}$ be the input set of m trajectories for which we want to compute its median trajectory τ^M . We define each trajectory in T as a list of at most n points, $\tau_i := (p_{i,1}, \dots, p_{i,k})$ where $1 \leq i \leq m$ and $2 \leq k \leq n$. Note that the number of points for each trajectory can be different. Every two consecutive points $p_{i,j}$ and $p_{i,j+1}$ ($1 \leq j \leq k-1$) are connected by a segment $s_{i,j} := (\overline{p_{i,j}, p_{i,j+1}})$.

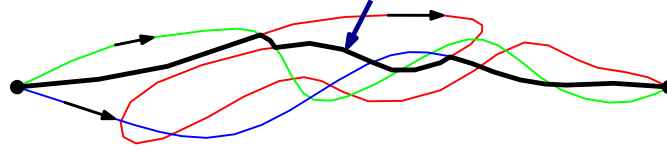
P is the set of all points in T , $P := \{p_{i,j} \mid i \in \{1 \dots m\}, j \in \{1 \dots n\}\}$ and S is the set of all segments in T , $S := \{s_{i,j} \mid i \in \{1 \dots m\}, j \in \{1 \dots n-1\}\}$. All trajectories in T share the same start and end points ($p_{1,1} = p_{2,1} = \dots = p_{m,1}$ and $p_{1,k_1} = p_{2,k_2} = \dots = p_{m,k_m}$ where $\{k_1, \dots, k_m\} \in \{1, \dots, n\}$).

Trajectories can intersect with other trajectories in other points than their start and end points. These intersection points are also included in the list of points that define the trajectory. When two segments intersect each other, then both segments will be split into two parts and all four segments share one intersection point as one of their endpoints (see Figure 2.1).

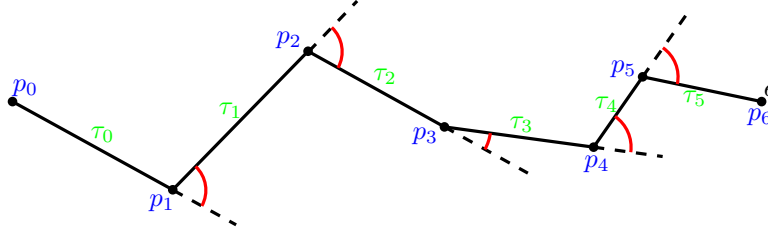
Let n' be the number of points in a trajectory, including their intersection points with other trajectories. In the worst case, $n' = mn^2$. In the rest of this thesis, we define n as a number of points in a trajectory, inclusive with its intersection points with other trajectories. Note that the number of segments for each trajectory is linear to the number of points, because trajectory with n points has $n-1$ segments.



Gambar 2.1: Numbering of points and segments



Gambar 2.2: Possible median trajectory (in black) with backward direction (indicated by the blue arrow)



Gambar 2.3: Red arcs indicate the angular change at each vertex

2.2 Properties of the Median Trajectory

We define several properties for the median trajectory τ^M with respect to the input set of trajectories T :

- τ^M is a directed polygonal line from start point to end point and should be similar to most trajectories in T .
- It must be built only using points and segments which are parts of trajectories in the input set.
- The usage of segments should follow the direction of them. Therefore, it is not allowed to use a segment such that the direction of τ^M is opposite to the direction of that segment in a trajectory (see Figure 2.2, indicated by the dark blue arrow).
- The length of the median trajectory should be relatively the same as the average length of all trajectories in the input set.
- The total angular change should also be similar to the average of total angular change of all trajectories in the input set. The total angular change of a trajectory is the sum of all angular changes at every vertex in that trajectory (see Figure 2.3).
- The number of vertices and edges of τ^M should be about the same with the average of the number of vertices and edges from all trajectories in the input set.

Using the definition of the input set of trajectories defined in the previous section, we define a median trajectory τ^M as a sequence of points from T , $\tau^M := (p_{i_1, j_1}, p_{i_2, j_2}, \dots, p_{i_k, j_k})$ where $\{i_1, i_2, \dots, i_k\} \in \{1 \dots m\}$ and $\{j_1, j_2, \dots, j_k\} \in \{1 \dots n\}$, or defined as a sequence of segments: $\tau^M := (s_{i_1, j_1}, s_{i_2, j_2}, \dots, s_{i_k, j_k})$ where $\{i_1, i_2, \dots, i_k\} \in \{1 \dots m\}$ and $\{j_1, j_2, \dots, j_k\} \in \{1 \dots n - 1\}$. Note that τ^M and all trajectories in T share the same start point and end point.

Table 2.1 shows how this information is kept in Γ .

Tabel 2.1: Table Γ after inserting \mathcal{S}_1

	v_{start}	\mathcal{S}_1	v_{end}
τ_1	1	12	20
τ_2	1		20
τ_3	1	9	20
τ_4	1		20

There are two possibilities of the placement of \mathcal{S}_2 :

Tabel 2.2: \mathcal{S}_2 between v_{start} and \mathcal{S}_1

	v_{start}	\mathcal{S}_2	\mathcal{S}_1	v_{end}
τ_1	1	5	12	20
τ_2	1	8		20
τ_3	1	2/8/17	9	20
τ_4	1			20

Tabel 2.3: \mathcal{S}_2 between \mathcal{S}_1 and v_{end}

	v_{start}	\mathcal{S}_1	\mathcal{S}_2	v_{end}
τ_1	1	12	5	20
τ_2	1		8	20
τ_3	1	9	2/8/17	20
τ_4	1			20

The final placement of table Γ after simplification:

Tabel 2.4: Final Γ

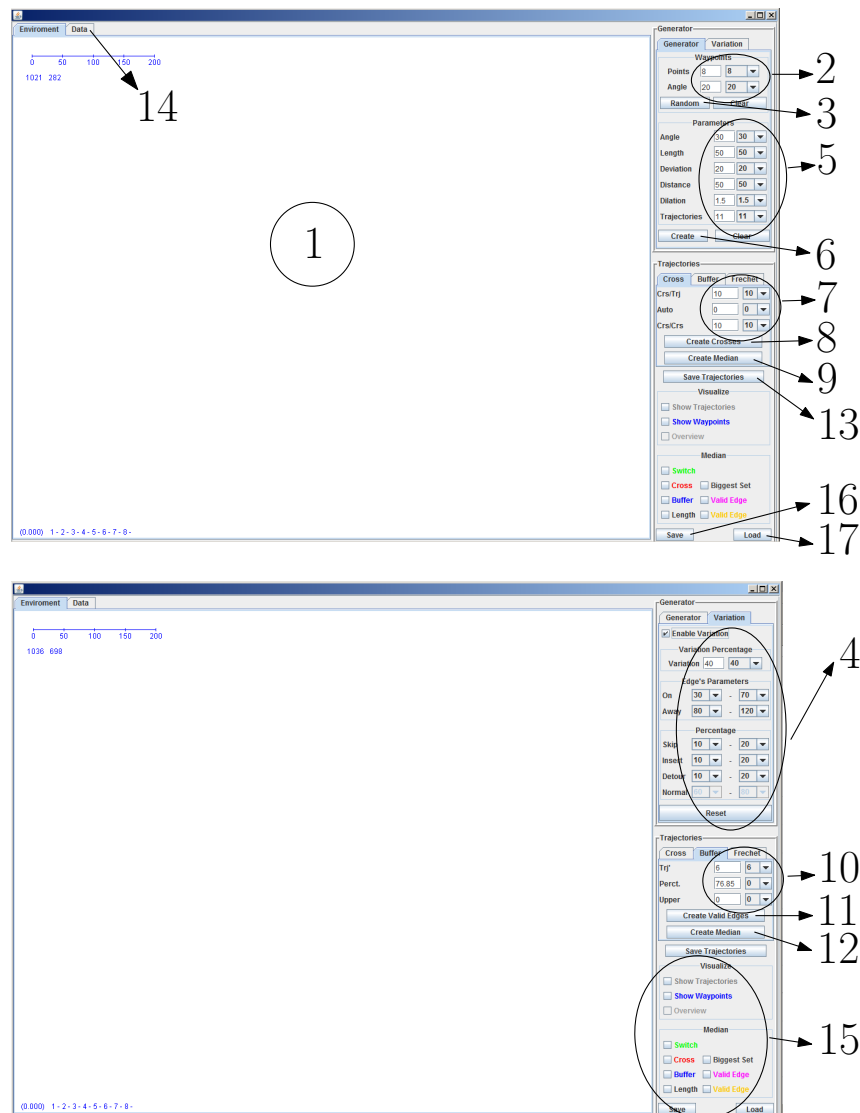
	v_{start}	\mathcal{S}_2	\mathcal{S}_1	v_{end}
τ_1	1	5	12	20
τ_2	1	8		20
τ_3	1	8	9	20
τ_4	1			20

DAFTAR REFERENSI

LAMPIRAN A

THE PROGRAM

The interface of the program is shown in Figure A.1:



Gambar A.1: Interface of the program

Step by step to compute the median trajectory using the program:

1. Create several waypoints. Click anywhere in the “Environment” area(1) or create them automatically by setting the parameters for waypoint(2) or clicking the button “Random”(3).

2. The “Variation” tab could be used to create variations by providing values needed to make them(4).
3. Create a set of trajectories by setting all parameters(5) and clicking the button “Create”(6).
4. Compute the median using the homotopic algorithm:
 - Define all parameters needed for the homotopic algorithm(7).
 - Create crosses by clicking the “Create Crosses” button(8).
 - Compute the median by clicking the “Compute Median” button(9).
5. Compute the median using the switching method and the buffer algorithm:
 - Define all parameters needed for the buffer algorithm(10).
 - Create valid edges by clicking the “Create Valid Edges”button(11).
 - Compute the median by clicking the “Compute Median”button(12).
6. Save the resulting median by clicking the “Save Trajectories” button(13). The result is saved in the computer memory and can be seen in “Data” tab(14)
7. The set of trajectories and its median trajectories will appear in the “Environment” area(1) and the user can change what to display by selecting various choices in “Visualize” and “Median” area(15).
8. To save all data to the disk, click the “Save”(16) button. A file dialog menu will appear.
9. To load data from the disk, click the “Load”(17) button.

LAMPIRAN B

THE SOURCE CODE

Listing B.1: MyFurSet.java

```

1
2 import java.util.ArrayList;
3 import java.util.Collections;
4 import java.util.HashSet;
5
6 /**
7  *
8  * @author Lionov
9  */
10
11 //class for set of vertices close to furthest edge
12 public class MyFurSet {
13     protected int id; //id of the set
14     protected MyEdge FurthestEdge; //the furthest edge
15     protected HashSet<MyVertex> set; //set of vertices close to furthest edge
16     protected ArrayList<ArrayList<Integer>> ordered; //list of all vertices in the set for each
17         trajectory
18     protected ArrayList<Integer> closeID; //store the ID of all vertices
19     protected ArrayList<Double> closeDist; //store the distance of all vertices
20     protected int totaltrj; //total trajectories in the set
21
22     /**
23      * Constructor
24      * @param id : id of the set
25      * @param totaltrj : total number of trajectories in the set
26      * @param FurthestEdge : the furthest edge
27      */
28     public MyFurSet(int id,int totaltrj,MyEdge FurthestEdge) {
29         this.id = id;
30         this.totaltrj = totaltrj;
31         this.FurthestEdge = FurthestEdge;
32         set = new HashSet<MyVertex>();
33         ordered = new ArrayList<ArrayList<Integer>>();
34         for (int i=0;i<totaltrj;i++) ordered.add(new ArrayList<Integer>());
35         closeID = new ArrayList<Integer>(totaltrj);
36         closeDist = new ArrayList<Double>(totaltrj);
37         for (int i = 0; i < totaltrj; i++) {
38             closeID.add(-1);
39             closeDist.add(Double.MAX_VALUE);
40         }
41     }
42
43     /**
44      * set a vertex into the set
45      * @param v : vertex to be added to the set
46      */
47     public void add(MyVertex v) {
48         set.add(v);
49     }
50
51     /**
52      * check whether vertex v is a member of the set
53      * @param v : vertex to be checked
54      * @return true if v is a member of the set , false otherwise
55      */
56     public boolean contains(MyVertex v) {
57         return this.set.contains(v);
58     }
59 }

```