LAPORAN TUGAS KECIL 2
IF2211 - STRATEGI ALGORITMA
IMPLEMENTASI ALGORITMA A* UNTUK MENENTUKAN LINTASAN TERPENDEK
SEMESTER II TAHUN 2020/2021

Oleh:
Aurelius Marcel Candra (13519198)

Prodi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung
2020

BAGIAN 1
SOURCE CODE

```
using System;
using System.Collections.Generic;
using System.Drawing;
using System.Linq;
using System.Windows.Forms;
using System.IO;

namespace A_Star
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e) // --> Event Button Browse diklik
        {
            // Browse Dialog, Inspirasi dari: https://www.c-sharpcorner.com/UploadFile/mahesh/openfiledialog-in-C-
Sharp/
            OpenFileDialog openFileDialographContainer = new OpenFileDialog
            {
                InitialDirectory = @"C:\",
                Title = "Browse Text Files",

                CheckFileExists = true,
                CheckPathExists = true,
                DefaultExt = "txt",
                Filter = "txt files (*.txt)|*.txt",
                FilterIndex = 2,
                RestoreDirectory = true,

                ReadOnlyChecked = true,
                ShowReadOnly = true
            };

            if (openFileDialographContainer.ShowDialog() == DialogResult.OK)    // --> Ketika Button OK pada Browse
diklik
            {
                // ### Section Clear, untuk Membuka File Baru
                richTextBox1.Clear();

                comboBox1.Text = "";
                comboBox1.Items.Clear();
                comboBox2.Text = "";
                comboBox2.Items.Clear();

                textBox2.Text = "";

                pictureBox1.Image = null;
                // ###

                textBox1.Text = openFileDialographContainer.FileName;

                string[] lines = File.ReadAllLines(openFileDialographContainer.FileName);

                int counter = 0;
                foreach (string line in lines)
                {
                    string[] temp = line.Split('\t');

                    if (counter == 0)   // --> Inisialisasi Komponen Global Kontainer
                    {
                        Global.graph = new (double, double)[temp.Count() - 1, temp.Count() - 1];
                        Global.nodes = new Dictionary<int, string>();
                        Global.nodes_inv = new Dictionary<string, int>();
                        Global.graph_Dict = new graphDictionary();

                        for (int i = 1; i < temp.Count(); i++)
                        {
                            Global.nodes.Add(i - 1, temp[i]);
                            Global.nodes_inv.Add(temp[i], i - 1);
                            comboBox1.Items.Add(temp[i]);
                        }
                    }

                    if (counter > 0)
                    {
                        for (int i = 1; i < temp.Count(); i++)
                        {
                            double dist_val = double.Parse(temp[i]);

                            Global.graph[counter - 1, i - 1] = (0.0, dist_val); // --> Value item1 = 0.0 sebagai
default, item2 = Isi dari File Adj Matrix

                            if (dist_val != 0)
                            {
                                Global.graph_Dict.AddEdge(Global.nodes[counter - 1], Global.nodes[i - 1]);
                                Global.graph_Dict.AddEdge(Global.nodes[i - 1], Global.nodes[counter - 1]);
                            }
                        }
                    }

                    counter++;
                }
            }
        }

        private void comboBox1_SelectedIndexChanged(object sender, EventArgs e) // --> Event Node pada ComboBox1 
dipilih
        {
            comboBox2.Items.Clear();
            comboBox2.Text = "";

            // Tambahkan Opsi Node ke ComboBox2 Selain Node yang Sudah Dipilih di ComboBox1
            foreach (string item in comboBox1.Items)
            {
                if (item != comboBox1.SelectedItem.ToString())
                {
                    comboBox2.Items.Add(item);
                }
            }
        }

        private void button3_Click(object sender, EventArgs e)  // --> Event Button Coords diklik
        {
            if (textBox1.Text == "")
            {
                richTextBox1.Text = "TIDAK ADA FILE YANG DIINPUTKAN, SILAKAN TEKAN BROWSE";
            }
            else
```

```
{
    OpenFileDialog openFileDialographContainer = new OpenFileDialog
    {
        InitialDirectory = @"C:\",
        Title = "Browse Text Files",

        CheckFileExists = true,
        CheckPathExists = true,

        DefaultExt = "txt",
        Filter = "txt files (*.txt)|*.txt",
        FilterIndex = 2,
        RestoreDirectory = true,

        ReadOnlyChecked = true,
        ShowReadOnly = true
    };

    if (openFileDialographContainer.ShowDialog() == DialogResult.OK)
    {
        textBox2.Text = openFileDialographContainer.FileName;

        string[] lines = File.ReadAllLines(openFileDialographContainer.FileName);

        List<(double, double)> coord_list = new List<(double, double)>();

        foreach (string line in lines)
        {
            string[] temp = line.Split(',');

            double d1 = double.Parse(temp[1]);
            double d2 = double.Parse(temp[2]);
```

```
                    coord_list.Add((d1, d2));
                }

                for (int i = 0; i < coord_list.Count; i++)
                {
                    for (int j = i; j < coord_list.Count; j++)
                    {
                        // ### Section Algoritma Haversine, Inspirasi dari:
https://gist.github.com/jammin77/033a332542aa24889452
                        double dLat = (coord_list[i].Item1 - coord_list[j].Item1) * (Math.PI / 180);
                        double dLon = (coord_list[i].Item2 - coord_list[j].Item2) * (Math.PI / 180);

                        double a = Math.Sin(dLat / 2) * Math.Sin(dLat / 2) +
                                   Math.Cos((coord_list[j].Item1) * (Math.PI / 180)) *
Math.Cos((coord_list[i].Item1) * (Math.PI / 180)) *
                                   Math.Sin(dLon / 2) * Math.Sin(dLon / 2);

                        double c = 2 * Math.Asin(Math.Min(1, Math.Sqrt(a)));

                        double d = 6371 * c;    // --> 6371 = default value untuk radius bumi dalam satuan km
                        // ###

                        Global.graph[i, j].Item1 = d;
                        Global.graph[j, i].Item1 = d;
                    }
                }

                // ### Section Print Graf Default
                Microsoft.Msagl.Drawing.Graph graph = new Microsoft.Msagl.Drawing.Graph("");

                for (int i = 0; i < Global.graph.GetLength(0); i++)
```

```
                    {
                        for (int j = i; j < Global.graph.GetLength(1); j++)
                        {
                            if (Global.graph[i, j].Item2 != 0.0)
                            {
                                graph.AddEdge(Global.nodes[i], Math.Round(Global.graph[i, j].Item1, 2).ToString(), 
Global.nodes[j]).Attr.ArrowheadAtTarget = Microsoft.Msagl.Drawing.ArrowStyle.None;
                            }
                        }
                    }

                    foreach (string node in Global.nodes_inv.Keys)
                    {
                        graph.FindNode(node).Attr.Shape = Microsoft.Msagl.Drawing.Shape.Circle;
                        graph.FindNode(node).Attr.FillColor = Microsoft.Msagl.Drawing.Color.Gray;
                    }

                    Microsoft.Msagl.GraphViewerGdi.GraphRenderer renderer = new 
Microsoft.Msagl.GraphViewerGdi.GraphRenderer(graph);

                    renderer.CalculateLayout();

                    int width = 267;

                    Bitmap bitmap = new Bitmap(width, (int)(graph.Height * (width / graph.Width)));

                    renderer.Render(bitmap);

                    pictureBox1.Image = bitmap;
                    // ###
                }
            }
```

```
        }

        private void button2_Click(object sender, EventArgs e)  // --> Event Button Submit diklik
        {
            richTextBox1.Clear();

            if (textBox1.Text == "")
            {
                richTextBox1.Text = "TIDAK ADA FILE YANG DIINPUTKAN, SILAKAN TEKAN BROWSE";
            }
            else if (textBox2.Text == "")
            {
                richTextBox1.Text = "TIDAK ADA FILE KOORDINAT YANG DIINPUTKAN, SILAKAN TEKAN COORDS";
            }
            else if (comboBox1.Text == "" || comboBox2.Text == "")
            {
                richTextBox1.Text = "TIDAK ADA NODE YANG TERPILIH, SILAKAN PILIH NODE PADA KEDUA FIELD
INTERSECTION";
            }
            else
            {
                string root = comboBox1.SelectedItem.ToString();
                string target = comboBox2.SelectedItem.ToString();

                List<(string, string)> searchOrder = new List<(string, string)>();  // --> Kontainer Tracking Edge
Pencarian yang Terbentuk

                Dictionary<string, bool> visited = new Dictionary<string, bool>();  // --> Kontainer Tracking Node
yang Sudah Dikunjungi

                foreach (string nodes in Global.nodes_inv.Keys)
                {
                    visited.Add(nodes, false);
                }

                List<(string, string)> edgeContainer = new List<(string, string)>();    // --> Kontainer Edge yang
Sudah Dikunjungi / Terbentuk

                List<(string, string, double, double, double)> candidate = new List<(string, string, double, double,
double)>();
                // --> Kontainer Himpunan Edge - Node yang Dapat Dikunjungi, Beserta Nilai Fungsi g(n) dan h(n)

                string cur_node = root;

                double gn = Global.graph[Global.nodes_inv[root], Global.nodes_inv[cur_node]].Item2;
                double hn = Global.graph[Global.nodes_inv[cur_node], Global.nodes_inv[target]].Item1;

                candidate.Add(("?", cur_node, gn, hn, gn + hn));    // --> Konsep Push / Enqueue Node Root

                bool target_found = false;  // --> Cek Route Terdefinisi atau Tidak

                if (checkBox1.Checked)
                {
                    richTextBox1.AppendText("Steps:\n");
                }

                while (!target_found && candidate.Count != 0)
                {
                    (string, string, double, double, double) node = candidate[0];
                    candidate.RemoveAt(0);  // --> Konsep Pop / Dequeue

                    cur_node = node.Item2;
                    visited[cur_node] = true;
                    gn = node.Item3;
```

```
edgeContainer.Add((node.Item1, node.Item2));

if (searchOrder.Count == 0) // --> Untuk Loop Pertama, Dapat Langsung Ditambahkan ke dalam List
{
    searchOrder.Add((node.Item1, node.Item2));
}
else
{   // Untuk Loop Berikutnya Perlu Pengecekan
    // Jika Kontinu (Edge Dapat Disambung dengan Elemen Terakhir Pencarian)
    if (searchOrder[searchOrder.Count - 1].Item2 == node.Item1)
    {
        searchOrder.Add((node.Item1, node.Item2));
    }
    // Jika Tidak Kontinu
    else
    {
        // ### Section Buat Baru List Sequence Pencarian
        List<(string, string)> sequence = new List<(string, string)>();
        sequence.Add((node.Item1, node.Item2));

        while (sequence[sequence.Count - 1].Item2 != root)
        {
            foreach ((string, string) edge in edgeContainer)
            {
                if (edge.Item2 == sequence[sequence.Count - 1].Item1)
                {
                    sequence.Add(edge);
                }
            }
        }
```

```
                    searchOrder.Clear();

                    sequence.Reverse();

                    foreach ((string, string) edge in sequence)
                    {
                        searchOrder.Add(edge);
                    }
                    // ###
                }
            }

            if (cur_node.Equals(target))
            {
                target_found = true;     // --> Break
            }

            // --> Melakukan Pencarian Node Baru (yang Belum Dikunjungi) Sesuai Dictionary string -> 
List<string>
            foreach (string node_adj in Global.graph_Dict.graph[cur_node])
            {
                if (!visited[node_adj])
                {
                    double new_gn = Global.graph[Global.nodes_inv[cur_node], 
Global.nodes_inv[node_adj]].Item2 + gn;
                    hn = Global.graph[Global.nodes_inv[node_adj], Global.nodes_inv[target]].Item1;
                    candidate.Add((node.Item2, node_adj, new_gn, hn, new_gn + hn));
                }
            }

            candidate.Sort((a, b) => a.Item5.CompareTo(b.Item5));
            // --> Sort Berdasarkan gn + hn, Dengan Jumlah Terkecil Pertama
```

```
        // --> Menyerupai Konsep Stack

        if (checkBox1.Checked)
        {
            foreach ((string, string) edge in searchOrder)
            {
                if (edge == searchOrder[searchOrder.Count - 1])
                {
                    richTextBox1.AppendText(edge.Item2 + "\n");
                }
                else
                {
                    richTextBox1.AppendText(edge.Item2 + " → ");
                }
            }
        }
    }

    if (checkBox1.Checked)
    {
        richTextBox1.AppendText("\n");
    }

    if (!target_found)
    {
        richTextBox1.AppendText("Route not available");
    }
    else
    {
        richTextBox1.AppendText("Final:\n");

        double distance = 0;
```

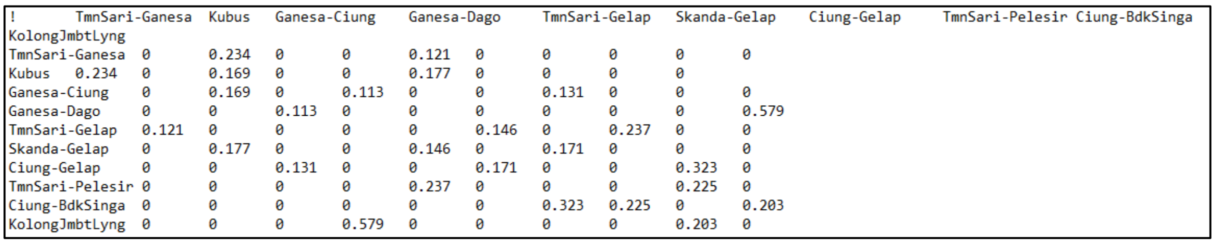
```
                    searchOrder.RemoveAt(0);

                    foreach ((string, string) el in searchOrder)
                    {
                        if (el == searchOrder[0])
                        {
                            richTextBox1.AppendText(el.Item1 + " → " + el.Item2);
                        }
                        else
                        {
                            richTextBox1.AppendText(" → " + el.Item2);
                        }

                        distance += Global.graph[Global.nodes_inv[el.Item1], Global.nodes_inv[el.Item2]].Item1;
                    }

                    richTextBox1.AppendText("\n\nDistance: " + Math.Round(distance, 2).ToString());

                    // ### Section Print Graf Baru
                    Microsoft.Msagl.Drawing.Graph graph = new Microsoft.Msagl.Drawing.Graph("");

                    for (int i = 0; i < Global.graph.GetLength(0); i++)
                    {
                        for (int j = i; j < Global.graph.GetLength(1); j++)
                        {
                            if (Global.graph[i, j].Item2 != 0)
                            {
                                if (searchOrder.Contains((Global.nodes[i], Global.nodes[j])))
                                {
                                    graph.AddEdge(Global.nodes[i], Math.Round(Global.graph[i, j].Item1,
2).ToString(), Global.nodes[j]).Attr.Color = Microsoft.Msagl.Drawing.Color.Red;
```

```
                                }
                                else if (searchOrder.Contains((Global.nodes[j], Global.nodes[i])))
                                {
                                    graph.AddEdge(Global.nodes[j], Math.Round(Global.graph[i, j].Item1,
2).ToString(), Global.nodes[i]).Attr.Color = Microsoft.Msagl.Drawing.Color.Red;
                                }
                                else
                                {
                                    graph.AddEdge(Global.nodes[i], Math.Round(Global.graph[i, j].Item1,
2).ToString(), Global.nodes[j]).Attr.ArrowheadAtTarget = Microsoft.Msagl.Drawing.ArrowStyle.None;
                                }
                            }
                        }
                    }

                    foreach (string node in Global.nodes_inv.Keys)
                    {
                        graph.FindNode(node).Attr.Shape = Microsoft.Msagl.Drawing.Shape.Circle;
                        graph.FindNode(node).Attr.FillColor = Microsoft.Msagl.Drawing.Color.Gray;
                    }

                    foreach ((string, string) edge in searchOrder)
                    {
                        graph.FindNode(edge.Item1).Attr.FillColor = Microsoft.Msagl.Drawing.Color.Red;
                        graph.FindNode(edge.Item2).Attr.FillColor = Microsoft.Msagl.Drawing.Color.Red;
                    }

                    Microsoft.Msagl.GraphViewerGdi.GraphRenderer renderer = new
Microsoft.Msagl.GraphViewerGdi.GraphRenderer(graph);

                    renderer.CalculateLayout();
                    int width = 267;

                    Bitmap bitmap = new Bitmap(width, (int)(graph.Height * (width / graph.Width)));

                    renderer.Render(bitmap);

                    pictureBox1.Image = bitmap;
                    // ###
                }
            }
        }
    }

    static class Global
    {
        public static (double, double)[,] graph;    // --> Kontainer Representasi Adj Matrix dengan value (h(n),
g(n))

        public static graphDictionary graph_Dict;   // --> Kontainer Representasi Graf Koneksi Antar Node

        public static Dictionary<int, string> nodes;    // --> Kontainer untuk Konversi Index -> Nama Node
(digunakan pada graph)

        public static Dictionary<string, int> nodes_inv;    // --> Kontainer untuk Konversi Nama Node -> Index
(digunakan pada graph)
    }

    class graphDictionary   // --> Kelas untuk menyimpan graf yang dalam bentuk dictionary (mapping string ke list
of string)
    {
        public Dictionary<string, List<string>> graph = new Dictionary<string, List<string>>();
        public void AddEdge(string v, string w)
        {
            if (graph.ContainsKey(v))
            {
                if (!graph[v].Contains(w))
                {
                    graph[v].Add(w);
                }
            }
            else
            {
                List<string> new_el = new List<string>();
                new_el.Add(w);

                graph.Add(v, new_el);
            }
        }
    }
}
```

BAGIAN 2
PETA / GRAF INPUT

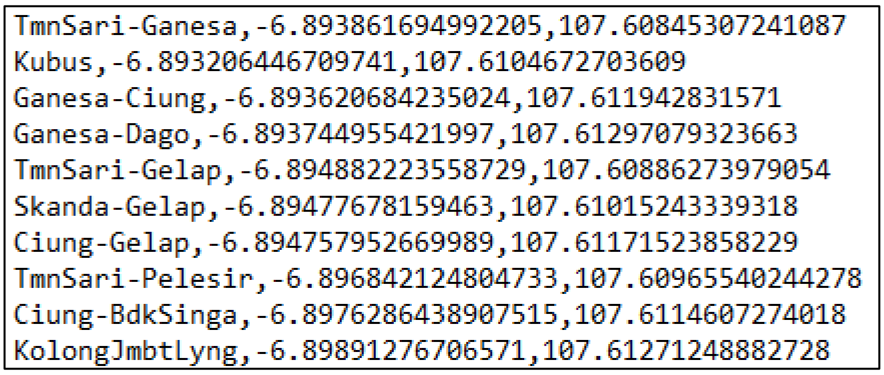
1. Wilayah ITB

- **Graph_ITB.txt**



```
!       TmnSari-Ganesa  Kubus   Ganesa-Ciung    Ganesa-Dago     TmnSari-Gelap   Skanda-Gelap    Ciung-Gelap     TmnSari-Pelesir Ciung-BdkSinga
KolongJmbtLyng
TmnSari-Ganesa  0       0.234   0       0       0.121   0       0       0       0       0
Kubus   0.234   0       0.169   0       0       0.177   0       0       0       0
Ganesa-Ciung    0       0.169   0       0.113   0       0       0.131   0       0       0
Ganesa-Dago     0       0       0.113   0       0       0       0       0       0       0.579
TmnSari-Gelap   0.121   0       0       0       0       0.146   0       0.237   0       0
Skanda-Gelap    0       0.177   0       0       0.146   0       0.171   0       0       0
Ciung-Gelap     0       0       0.131   0       0       0.171   0       0       0.323   0
TmnSari-Pelesir 0       0       0       0       0.237   0       0       0       0.225   0
Ciung-BdkSinga  0       0       0       0       0       0       0.323   0.225   0       0.203
KolongJmbtLyng  0       0       0       0.579   0       0       0       0       0.203   0
```

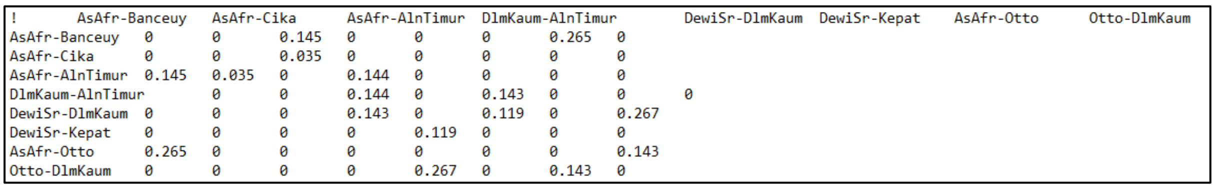
- **Coords_ITB.txt**



```
TmnSari-Ganesa,-6.893861694992205,107.60845307241087
Kubus,-6.893206446709741,107.6104672703609
Ganesa-Ciung,-6.893620684235024,107.611942831571
Ganesa-Dago,-6.893744955421997,107.61297079323663
TmnSari-Gelap,-6.894882223558729,107.60886273979054
Skanda-Gelap,-6.89477678159463,107.61015243339318
Ciung-Gelap,-6.894757952669989,107.61171523858229
TmnSari-Pelesir,-6.896842124804733,107.60965540244278
Ciung-BdkSinga,-6.8976286438907515,107.6114607274018
KolongJmbtLyng,-6.89891276706571,107.61271248882728
```

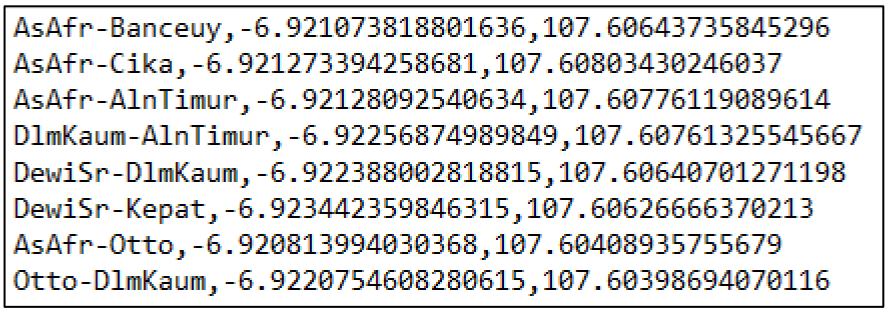
2. Wilayah Alun-Alun

- **Graph_Alun2.txt**



```
!       AsAfr-Banceuy   AsAfr-Cika      AsAfr-AlnTimur  DlmKaum-AlnTimur        DewiSr-DlmKaum  DewiSr-Kepat    AsAfr-Otto      Otto-DlmKaum
AsAfr-Banceuy   0       0       0.145   0       0       0       0.265   0
AsAfr-Cika      0       0       0.035   0       0       0       0       0
AsAfr-AlnTimur  0.145   0.035   0       0.144   0       0       0       0
DlmKaum-AlnTimur        0       0       0.144   0       0.143   0       0       0
DewiSr-DlmKaum  0       0       0       0.143   0       0.119   0       0.267
DewiSr-Kepat    0       0       0       0       0.119   0       0       0
AsAfr-Otto      0.265   0       0       0       0       0       0       0.143
Otto-DlmKaum    0       0       0       0       0.267   0       0.143   0
```

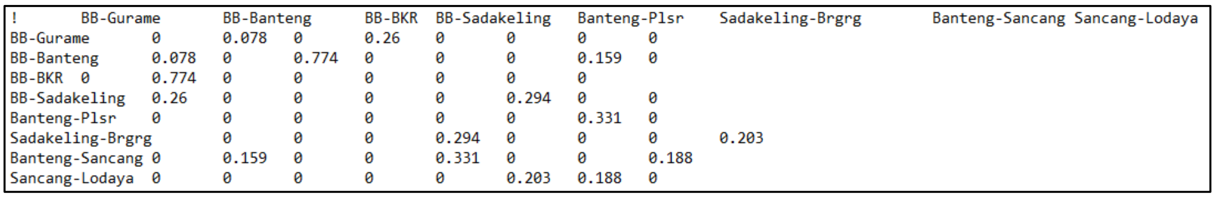
- **Coords_Alun2.txt**



```
AsAfr-Banceuy,-6.921073818801636,107.60643735845296
AsAfr-Cika,-6.921273394258681,107.60803430246037
AsAfr-AlnTimur,-6.92128092540634,107.60776119089614
DlmKaum-AlnTimur,-6.92256874989849,107.60761325545667
DewiSr-DlmKaum,-6.922388002818815,107.60640701271198
DewiSr-Kepat,-6.923442359846315,107.60626666370213
AsAfr-Otto,-6.920813994030368,107.60408935755679
Otto-DlmKaum,-6.9220754608280615,107.60398694070116
```

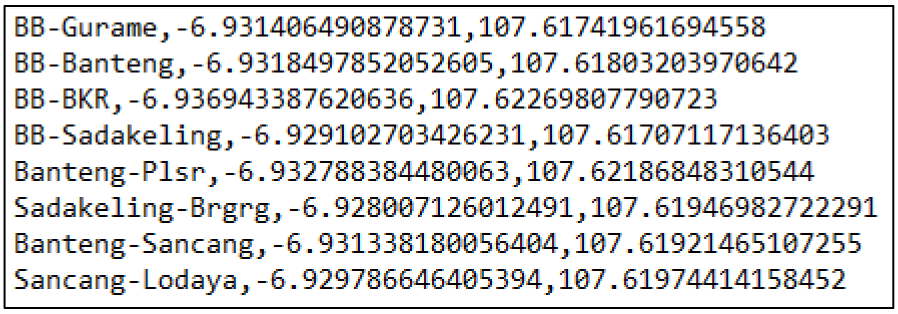
3. **Wilayah Buah Batu**

- **Graph_BuahBatu.txt**



```
!       BB-Gurame       BB-Banteng      BB-BKR  BB-Sadakeling   Banteng-Plsr    Sadakeling-Brgrg        Banteng-Sancang Sancang-Lodaya
BB-Gurame       0       0.078   0       0.26    0       0       0       0
BB-Banteng      0.078   0       0.774   0       0       0       0.159   0
BB-BKR  0       0.774   0       0       0       0       0       0
BB-Sadakeling   0.26    0       0       0       0       0.294   0       0
Banteng-Plsr    0       0       0       0       0       0       0.331   0
Sadakeling-Brgrg        0       0       0       0.294   0       0       0       0.203
Banteng-Sancang 0       0.159   0       0       0.331   0       0       0.188
Sancang-Lodaya  0       0       0       0       0       0.203   0.188   0
```

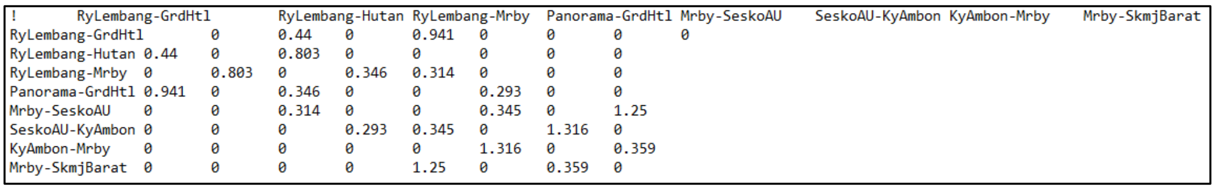
- **Coords_BuahBatu.txt**



```
BB-Gurame,-6.931406490878731,107.61741961694558
BB-Banteng,-6.9318497852052605,107.61803203970642
BB-BKR,-6.936943387620636,107.62269807790723
BB-Sadakeling,-6.929102703426231,107.61707117136403
Banteng-Plsr,-6.932788384480063,107.62186848310544
Sadakeling-Brgrg,-6.928007126012491,107.61946982722291
Banteng-Sancang,-6.931338180056404,107.61921465107255
Sancang-Lodaya,-6.929786646405394,107.61974414158452
```

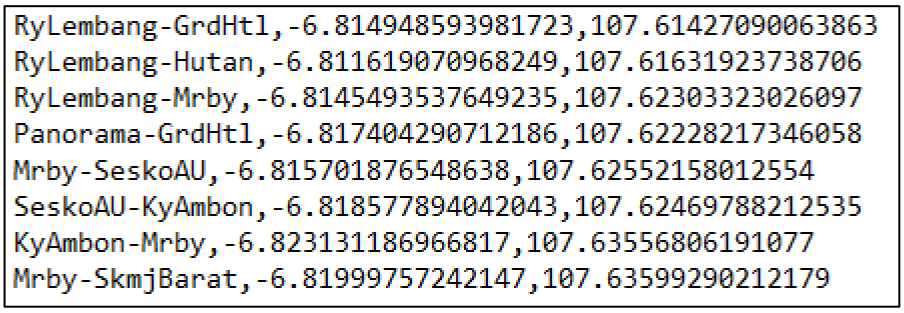
4. **Wilayah Lembang**
   *) Lembang dipilih karena wilayah tempat tinggal sudah terdapat pada wilayah Alun-Alun

- **Graph_Lembang.txt**



```
!       RyLembang-GrdHtl        RyLembang-Hutan RyLembang-Mrby  Panorama-GrdHtl Mrby-SeskoAU    SeskoAU-KyAmbon KyAmbon-Mrby    Mrby-SkmjBarat
RyLembang-GrdHtl        0       0.44    0       0.941   0       0       0       0
RyLembang-Hutan 0.44    0       0.803   0       0       0       0       0
RyLembang-Mrby  0       0.803   0       0.346   0.314   0       0       0
Panorama-GrdHtl 0.941   0       0.346   0       0       0.293   0       0
Mrby-SeskoAU    0       0       0.314   0       0       0.345   0       1.25
SeskoAU-KyAmbon 0       0       0       0.293   0.345   0       1.316   0
KyAmbon-Mrby    0       0       0       0       0       1.316   0       0.359
Mrby-SkmjBarat  0       0       0       0       1.25    0       0.359   0
```

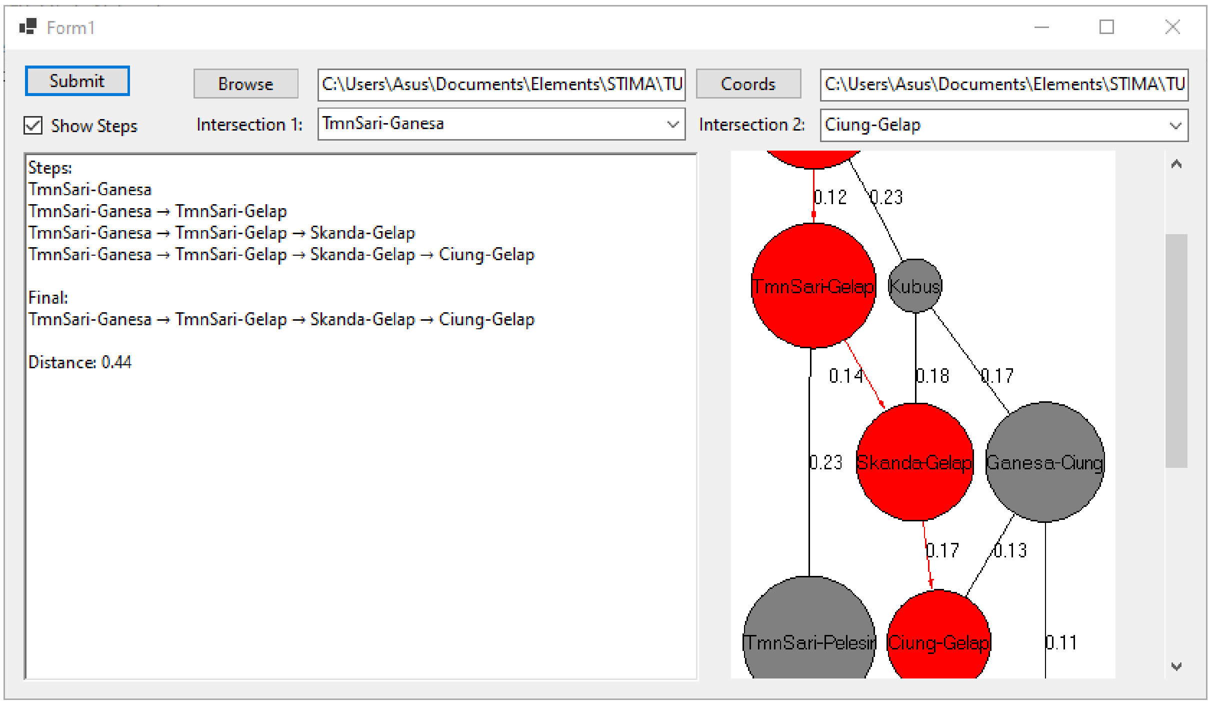
- **Coords_Lembang.txt**



```
RyLembang-GrdHtl,-6.814948593981723,107.61427090063863
RyLembang-Hutan,-6.811619070968249,107.61631923738706
RyLembang-Mrby,-6.8145493537649235,107.62303323026097
Panorama-GrdHtl,-6.817404290712186,107.62228217346058
Mrby-SeskoAU,-6.815701876548638,107.62552158012554
SeskoAU-KyAmbon,-6.818577894042043,107.62469788212535
KyAmbon-Mrby,-6.823131186966817,107.63556806191077
Mrby-SkmjBarat,-6.81999757242147,107.63599290212179
```

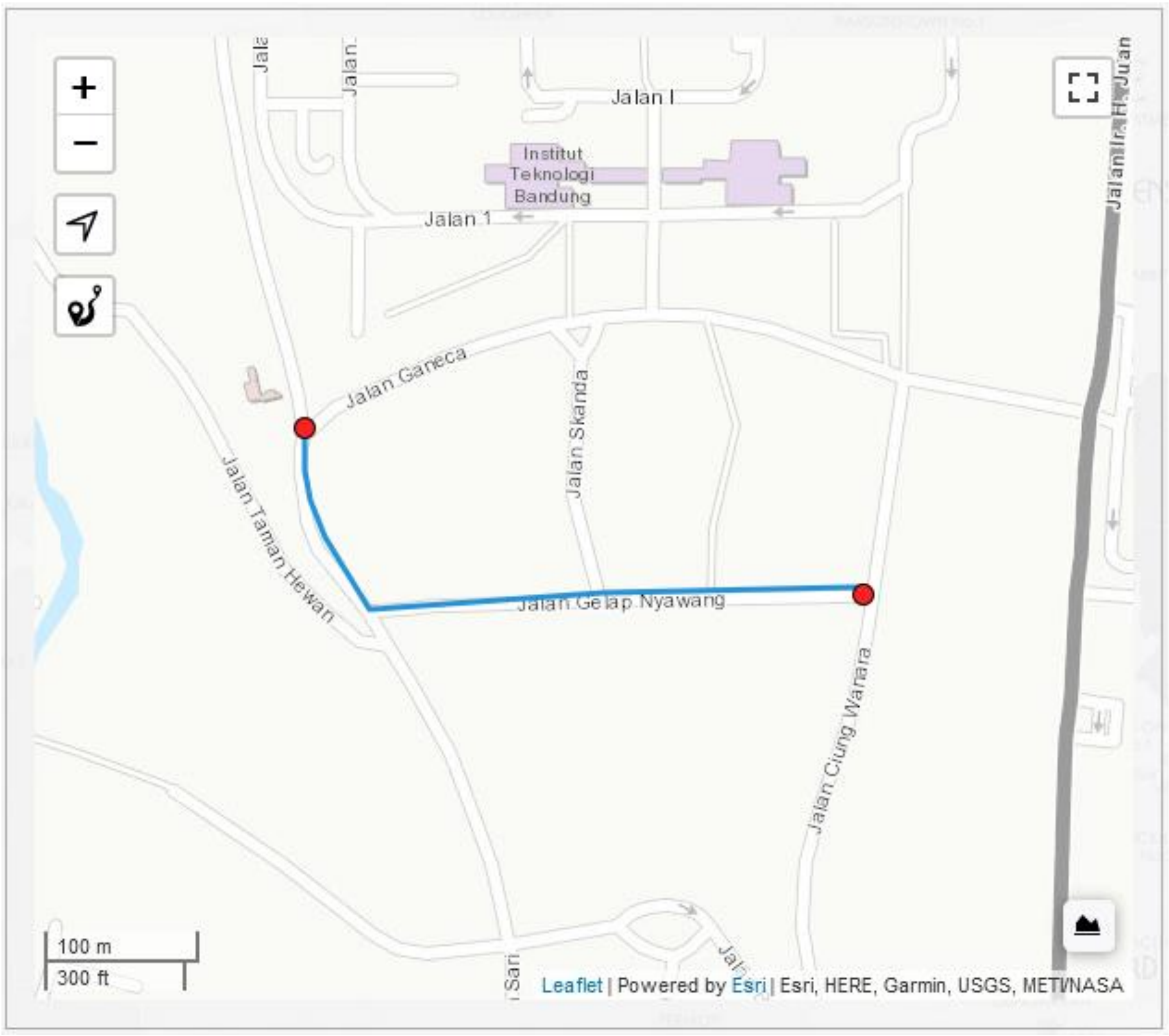
BAGIAN 3
SCREENSHOT HASIL DAN PETA

1. **Wilayah ITB**
- **Hasil Program:**

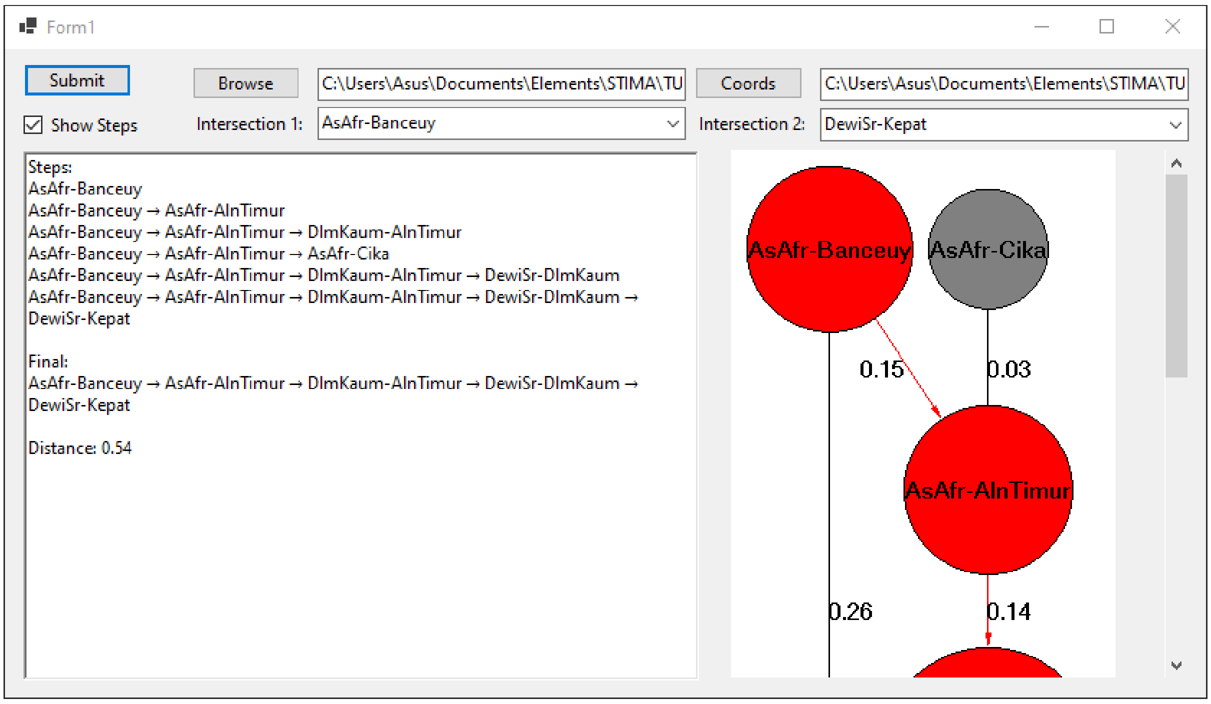


```
Steps:
TmnSari-Ganesa
TmnSari-Ganesa → TmnSari-Gelap
TmnSari-Ganesa → TmnSari-Gelap → Skanda-Gelap
TmnSari-Ganesa → TmnSari-Gelap → Skanda-Gelap → Ciung-Gelap

Final:
TmnSari-Ganesa → TmnSari-Gelap → Skanda-Gelap → Ciung-Gelap

Distance: 0.44
```

- **Hasil Peta:**


Jalan I
Institut Teknologi Bandung
Jalan 1
Jalan Ganeca
Jalan Skanda
Jalan Taman Hewan
Jalan Gelap Nyawang
Jalan Ciung Wanara
Jalan Ir. H. Juan
100 m
300 ft
Leaflet | Powered by Esri | Esri, HERE, Garmin, USGS, METI/NASA

2. **Wilayah Alun-Alun**
- **Hasil Program:**



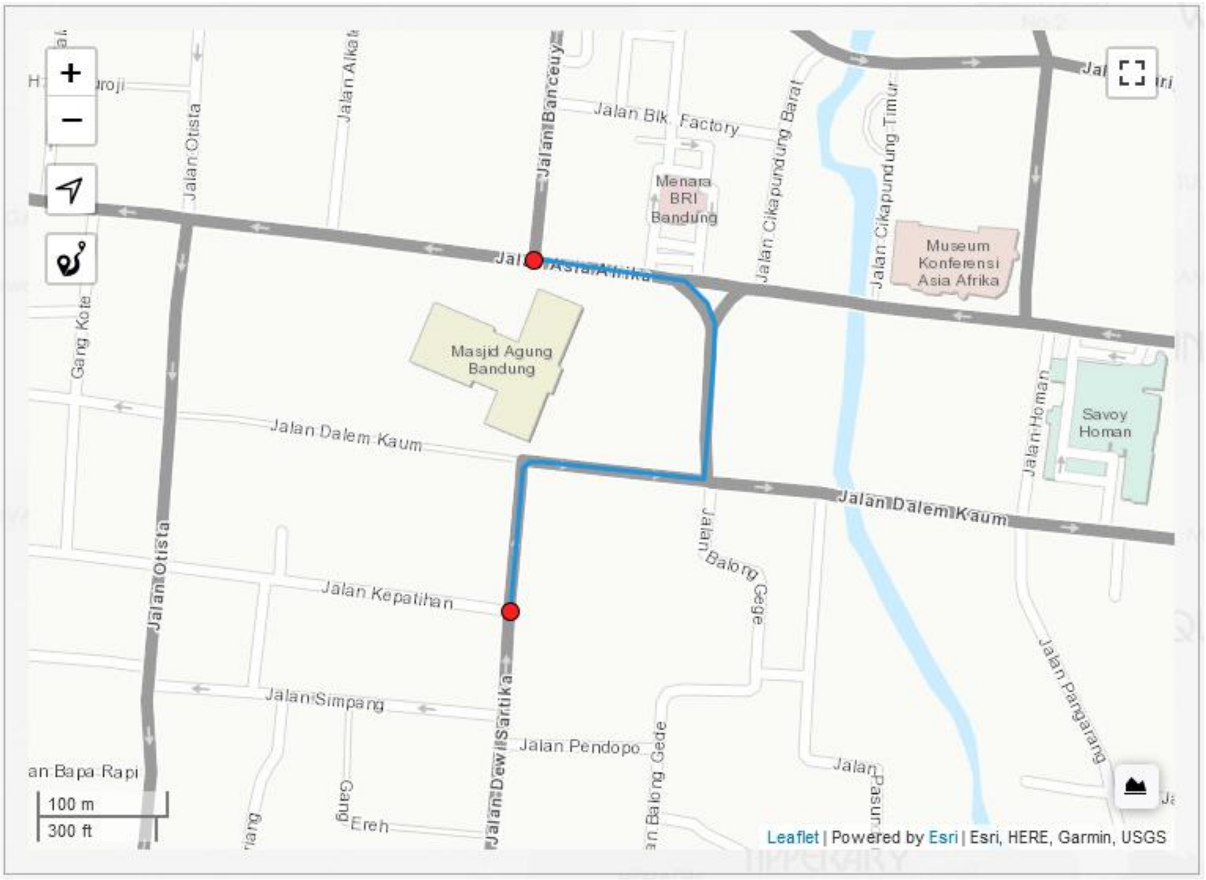
C:\Users\Asus\Documents\Elements\STIMA\TU

Intersection 1: AsAfr-Banceuy

Intersection 2: DewiSr-Kepat

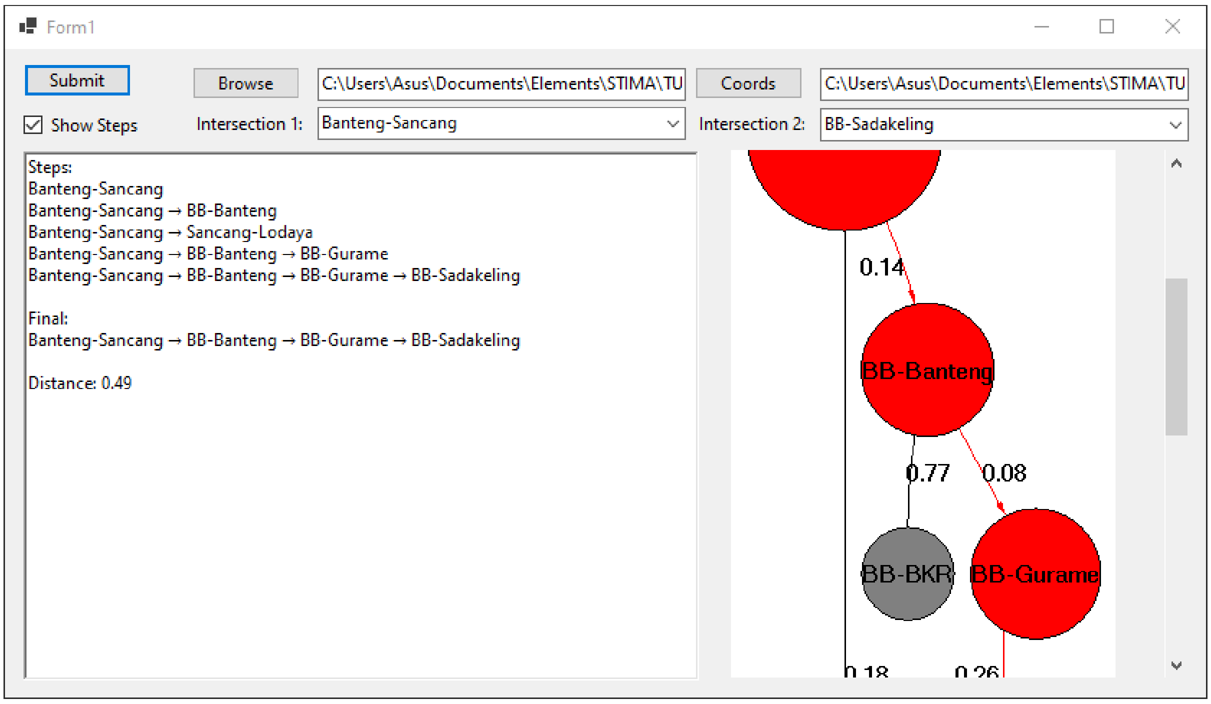
```
Steps:
AsAfr-Banceuy
AsAfr-Banceuy → AsAfr-AlnTimur
AsAfr-Banceuy → AsAfr-AlnTimur → DlmKaum-AlnTimur
AsAfr-Banceuy → AsAfr-AlnTimur → AsAfr-Cika
AsAfr-Banceuy → AsAfr-AlnTimur → DlmKaum-AlnTimur → DewiSr-DlmKaum
AsAfr-Banceuy → AsAfr-AlnTimur → DlmKaum-AlnTimur → DewiSr-DlmKaum →
DewiSr-Kepat

Final:
AsAfr-Banceuy → AsAfr-AlnTimur → DlmKaum-AlnTimur → DewiSr-DlmKaum →
DewiSr-Kepat

Distance: 0.54
```

- **Hasil Peta:**



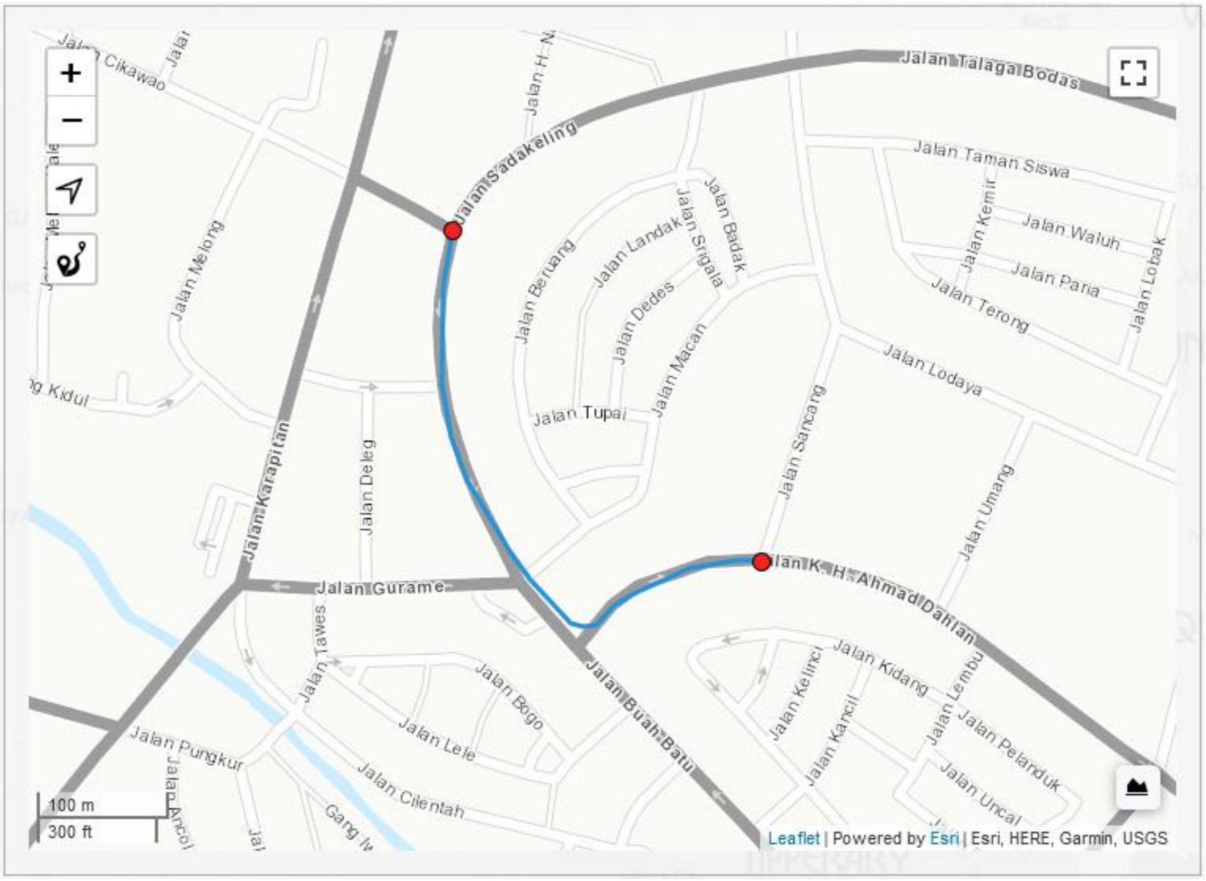
3. **Wilayah Buah Batu**

- **Hasil Program:**



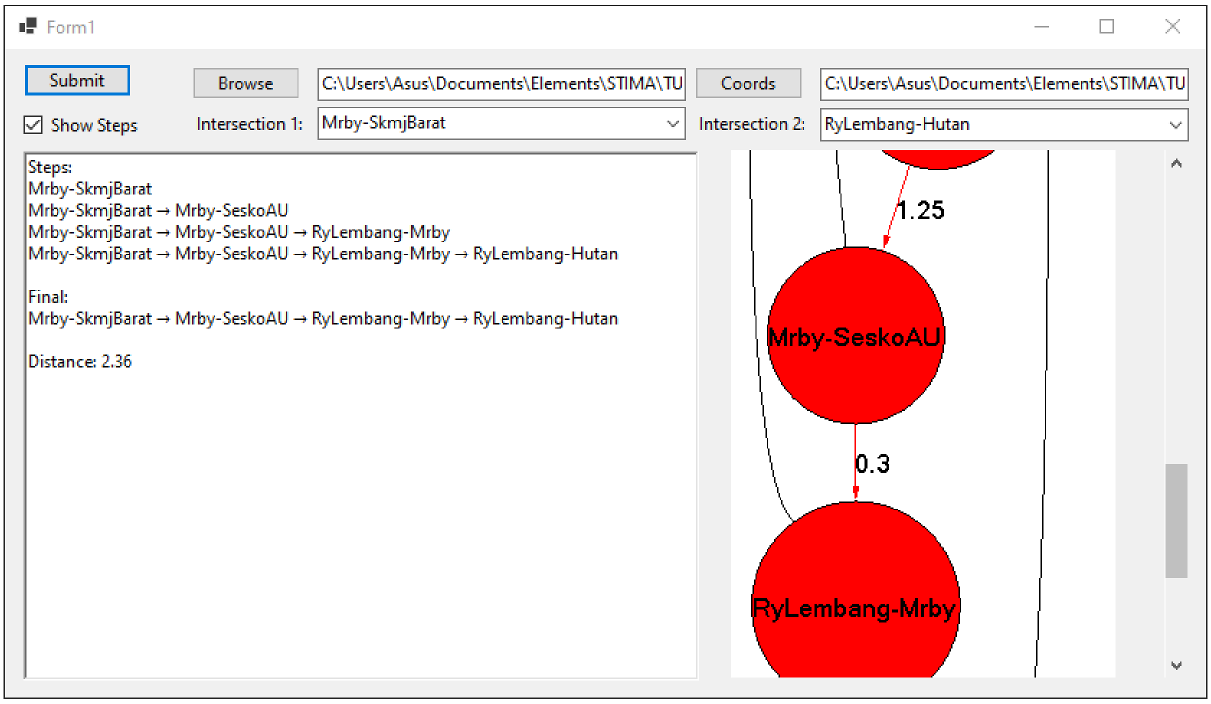
```
Steps:
Banteng-Sancang
Banteng-Sancang → BB-Banteng
Banteng-Sancang → Sancang-Lodaya
Banteng-Sancang → BB-Banteng → BB-Gurame
Banteng-Sancang → BB-Banteng → BB-Gurame → BB-Sadakeling

Final:
Banteng-Sancang → BB-Banteng → BB-Gurame → BB-Sadakeling

Distance: 0.49
```

- **Hasil Peta:**



4. **Wilayah Lembang**

- **Hasil Program:**



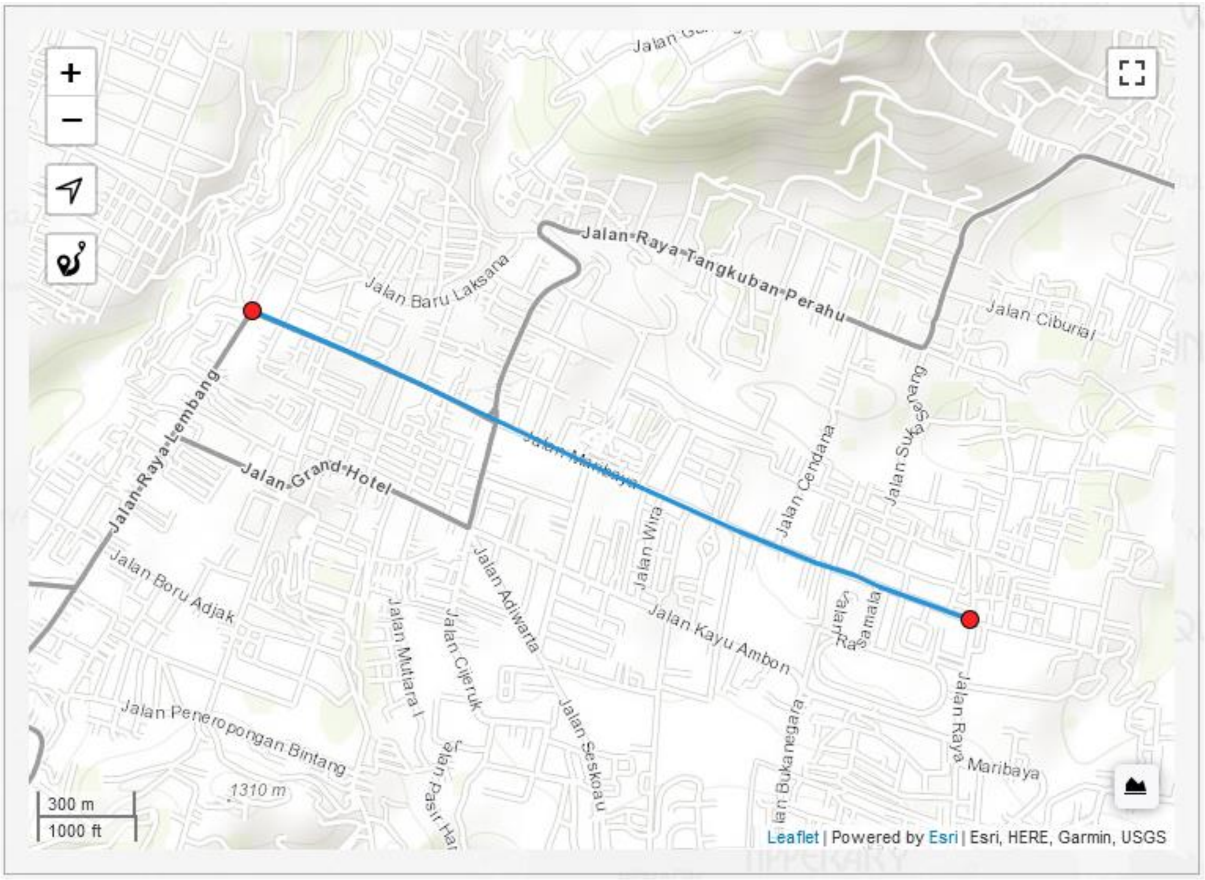
Form1

Submit | Browse | C:\Users\Asus\Documents\Elements\STIMA\TU | Coords | C:\Users\Asus\Documents\Elements\STIMA\TU

☑ Show Steps | Intersection 1: Mrby-SkmjBarat | Intersection 2: RyLembang-Hutan

```
Steps:
Mrby-SkmjBarat
Mrby-SkmjBarat → Mrby-SeskoAU
Mrby-SkmjBarat → Mrby-SeskoAU → RyLembang-Mrby
Mrby-SkmjBarat → Mrby-SeskoAU → RyLembang-Mrby → RyLembang-Hutan

Final:
Mrby-SkmjBarat → Mrby-SeskoAU → RyLembang-Mrby → RyLembang-Hutan

Distance: 2.36
```

- **Hasil Peta:**



LINK SOURCE CODE: https://github.com/Ardovigus/TUCIL3_IF2211_13519198

Poin	Ya	Tidak
1. Program dapat menerima input graf	✓	
2. Program dapat menghitung lintasan terpendek	✓	
3. Program dapat menampilkan lintasan terpendek serta jaraknya	✓	
4. Bonus: Program dapat menerima input peta dengan Google Map API dan menampilkan peta		✓