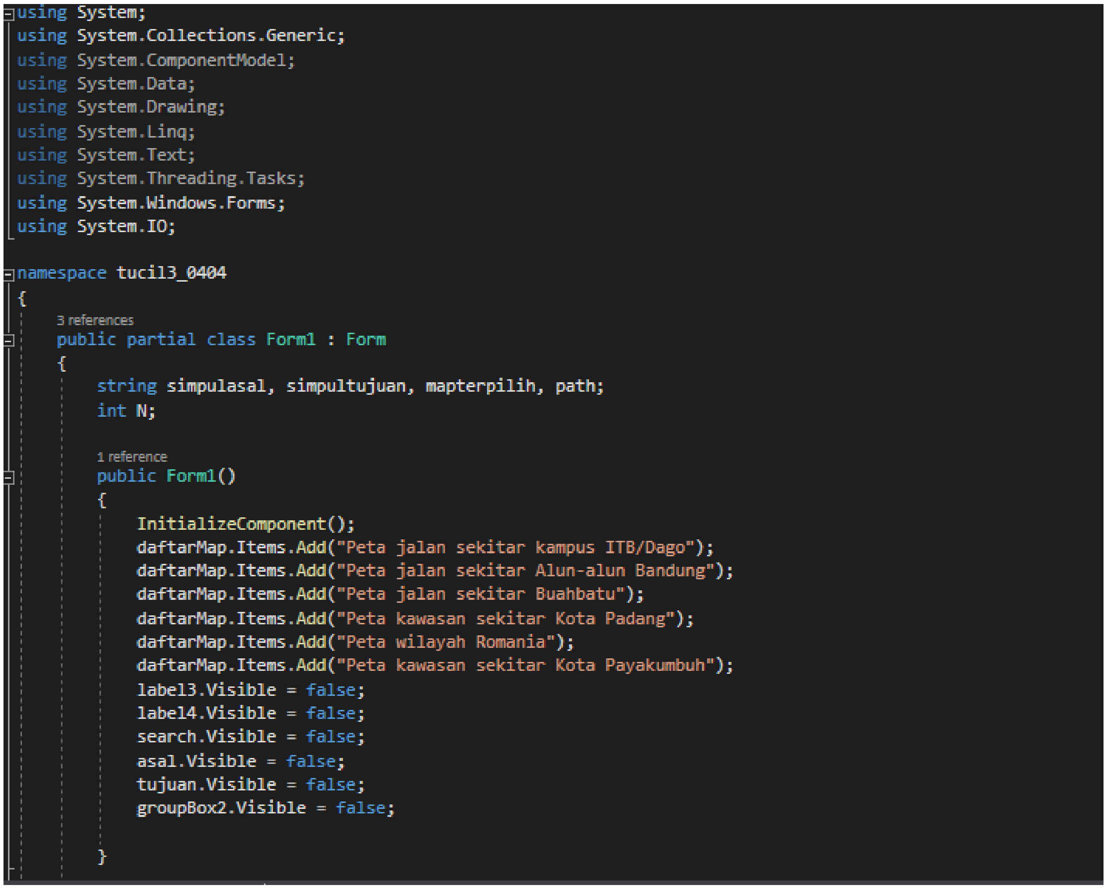
LAPORAN TUGAS KECIL 3
IF2211 STRATEGI ALGORITMA
Implementasi Algoritma A* untuk Menentukan Lintasan Terpendek

Disusun Oleh
Ruhiyah Faradishi Widiaputri 13519034 (K01)
Nabila Hannania 13519097 (K02)

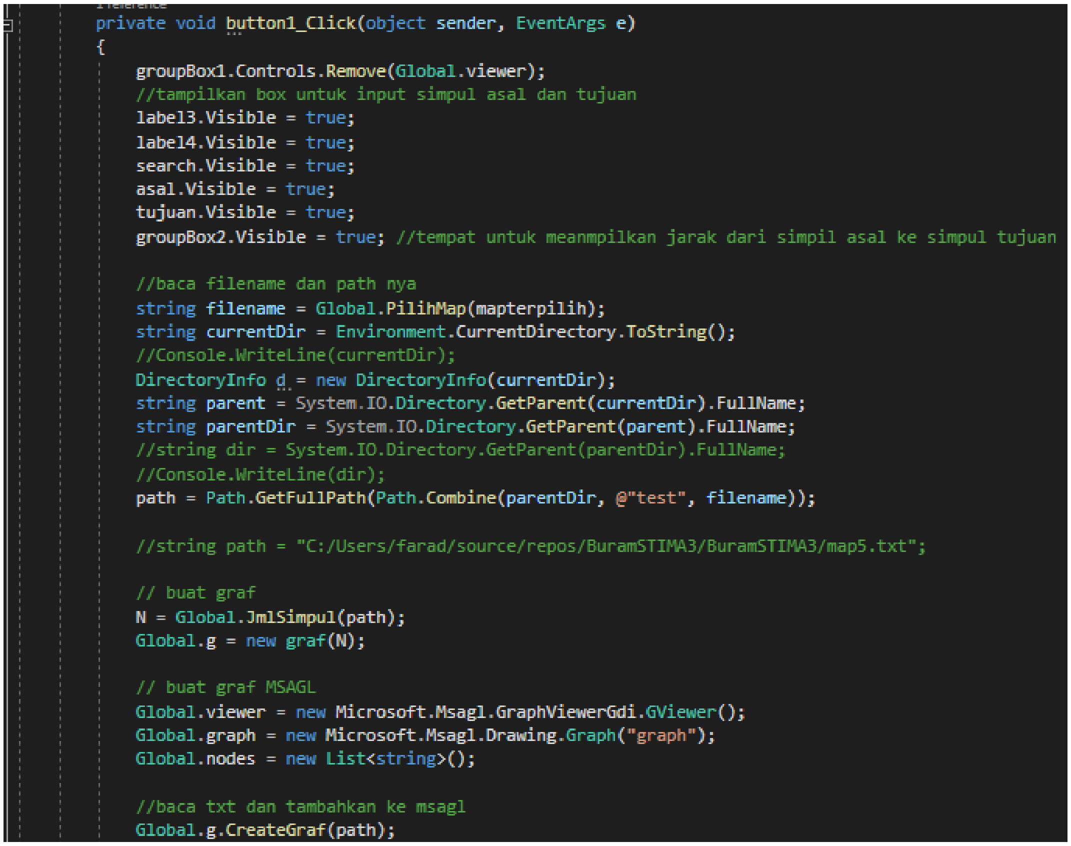
SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA
INSTITUT TEKNOLOGI BANDUNG
2021

A. Kode Program

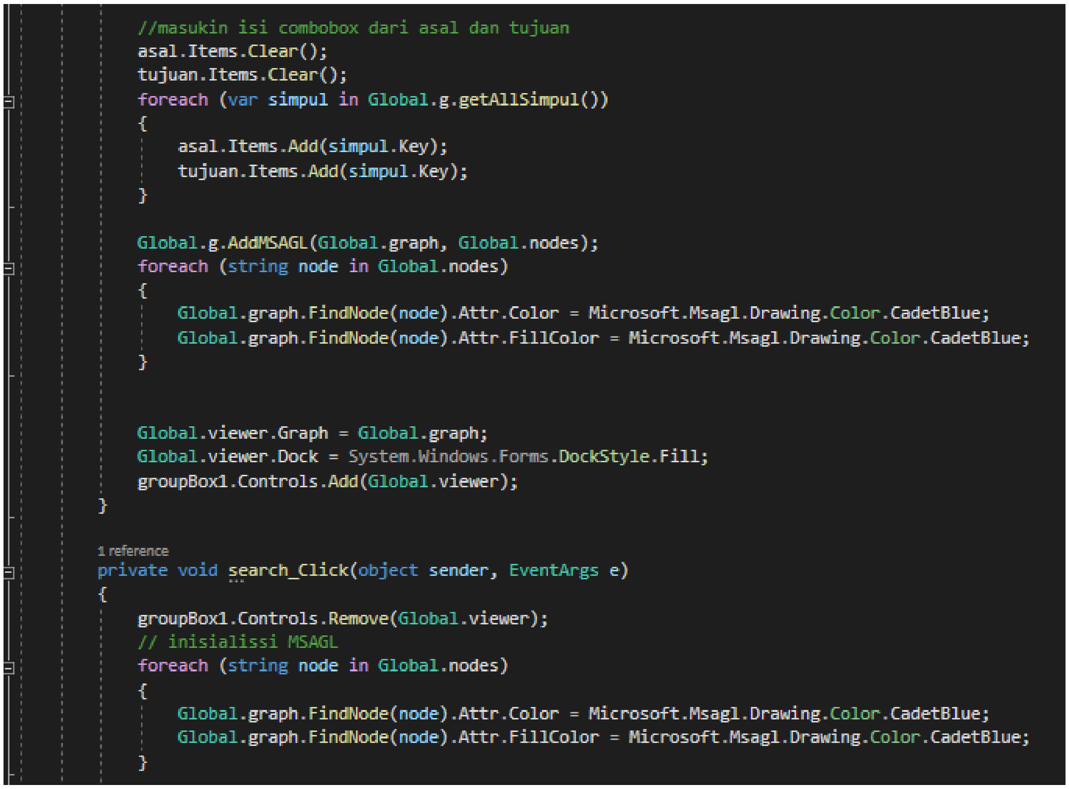
1. File Form1,cs



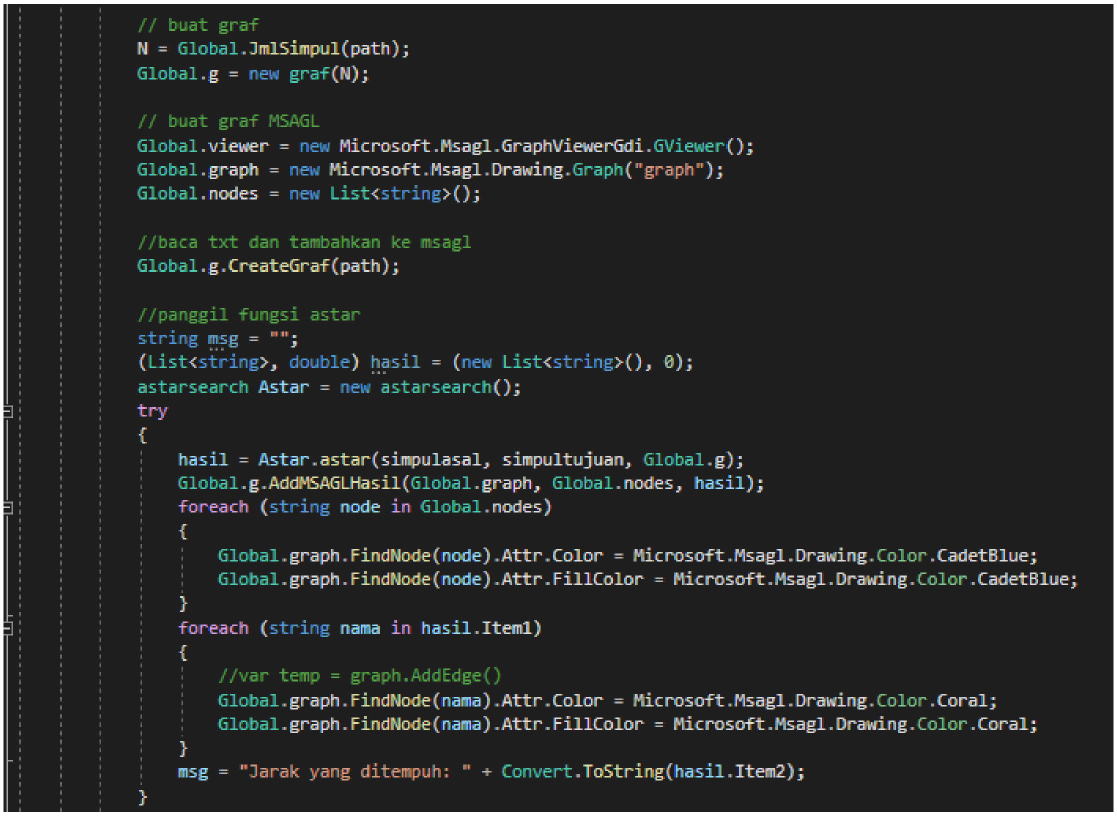
```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.IO;

namespace tucil3_0404
{
    3 references
    public partial class Form1 : Form
    {
        string simpulasal, simpultujuan, mapterpilih, path;
        int N;

        1 reference
        public Form1()
        {
            InitializeComponent();
            daftarMap.Items.Add("Peta jalan sekitar kampus ITB/Dago");
            daftarMap.Items.Add("Peta jalan sekitar Alun-alun Bandung");
            daftarMap.Items.Add("Peta jalan sekitar Buahbatu");
            daftarMap.Items.Add("Peta kawasan sekitar Kota Padang");
            daftarMap.Items.Add("Peta wilayah Romania");
            daftarMap.Items.Add("Peta kawasan sekitar Kota Payakumbuh");
            label3.Visible = false;
            label4.Visible = false;
            search.Visible = false;
            asal.Visible = false;
            tujuan.Visible = false;
            groupBox2.Visible = false;

        }
```



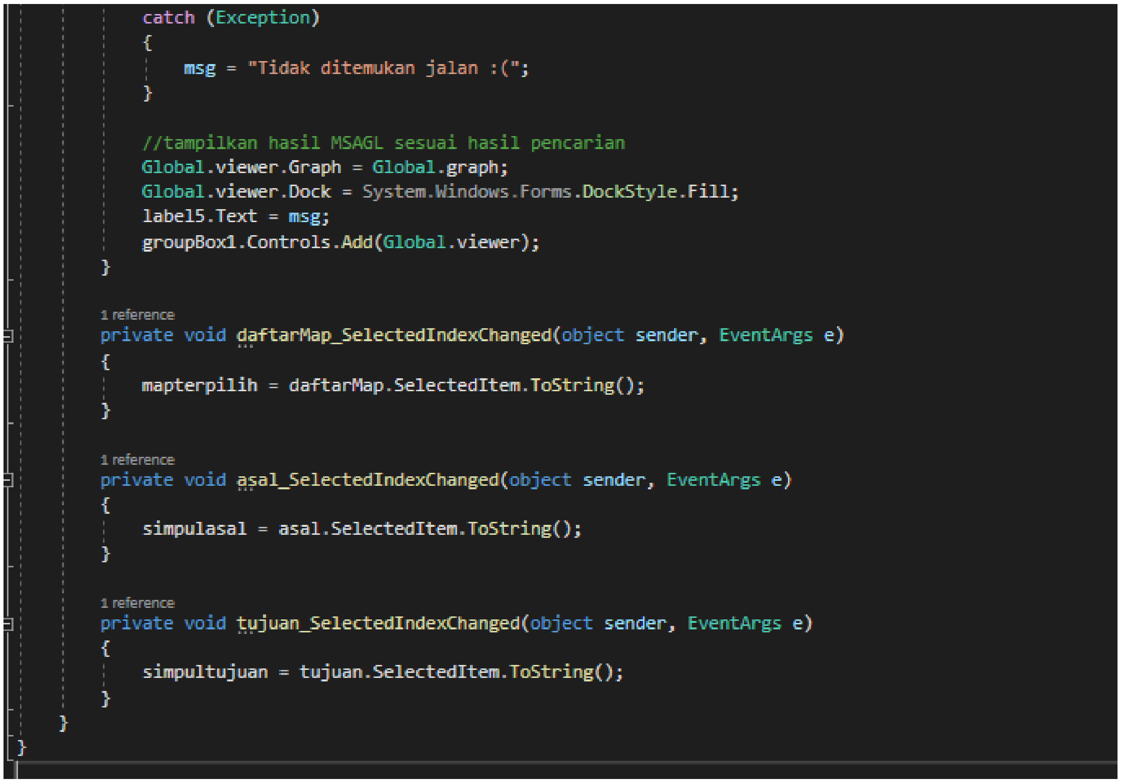
```
private void button1_Click(object sender, EventArgs e)
{
    groupBox1.Controls.Remove(Global.viewer);
    //tampilkan box untuk input simpul asal dan tujuan
    label3.Visible = true;
    label4.Visible = true;
    search.Visible = true;
    asal.Visible = true;
    tujuan.Visible = true;
    groupBox2.Visible = true; //tempat untuk meanmpilkan jarak dari simpil asal ke simpul tujuan

    //baca filename dan path nya
    string filename = Global.PilihMap(mapterpilih);
    string currentDir = Environment.CurrentDirectory.ToString();
    //Console.WriteLine(currentDir);
    DirectoryInfo d = new DirectoryInfo(currentDir);
    string parent = System.IO.Directory.GetParent(currentDir).FullName;
    string parentDir = System.IO.Directory.GetParent(parent).FullName;
    //string dir = System.IO.Directory.GetParent(parentDir).FullName;
    //Console.WriteLine(dir);
    path = Path.GetFullPath(Path.Combine(parentDir, @"test", filename));

    //string path = "C:/Users/farad/source/repos/BuramSTIMA3/BuramSTIMA3/map5.txt";

    // buat graf
    N = Global.JmlSimpul(path);
    Global.g = new graf(N);

    // buat graf MSAGL
    Global.viewer = new Microsoft.Msagl.GraphViewerGdi.GViewer();
    Global.graph = new Microsoft.Msagl.Drawing.Graph("graph");
    Global.nodes = new List<string>();

    //baca txt dan tambahkan ke msagl
    Global.g.CreateGraf(path);
```



```
    //masukin isi combobox dari asal dan tujuan
    asal.Items.Clear();
    tujuan.Items.Clear();
    foreach (var simpul in Global.g.getAllSimpul())
    {
        asal.Items.Add(simpul.Key);
        tujuan.Items.Add(simpul.Key);
    }

    Global.g.AddMSAGL(Global.graph, Global.nodes);
    foreach (string node in Global.nodes)
    {
        Global.graph.FindNode(node).Attr.Color = Microsoft.Msagl.Drawing.Color.CadetBlue;
        Global.graph.FindNode(node).Attr.FillColor = Microsoft.Msagl.Drawing.Color.CadetBlue;
    }


    Global.viewer.Graph = Global.graph;
    Global.viewer.Dock = System.Windows.Forms.DockStyle.Fill;
    groupBox1.Controls.Add(Global.viewer);
}

1 reference
private void search_Click(object sender, EventArgs e)
{
    groupBox1.Controls.Remove(Global.viewer);
    // inisialissi MSAGL
    foreach (string node in Global.nodes)
    {
        Global.graph.FindNode(node).Attr.Color = Microsoft.Msagl.Drawing.Color.CadetBlue;
        Global.graph.FindNode(node).Attr.FillColor = Microsoft.Msagl.Drawing.Color.CadetBlue;
    }
```

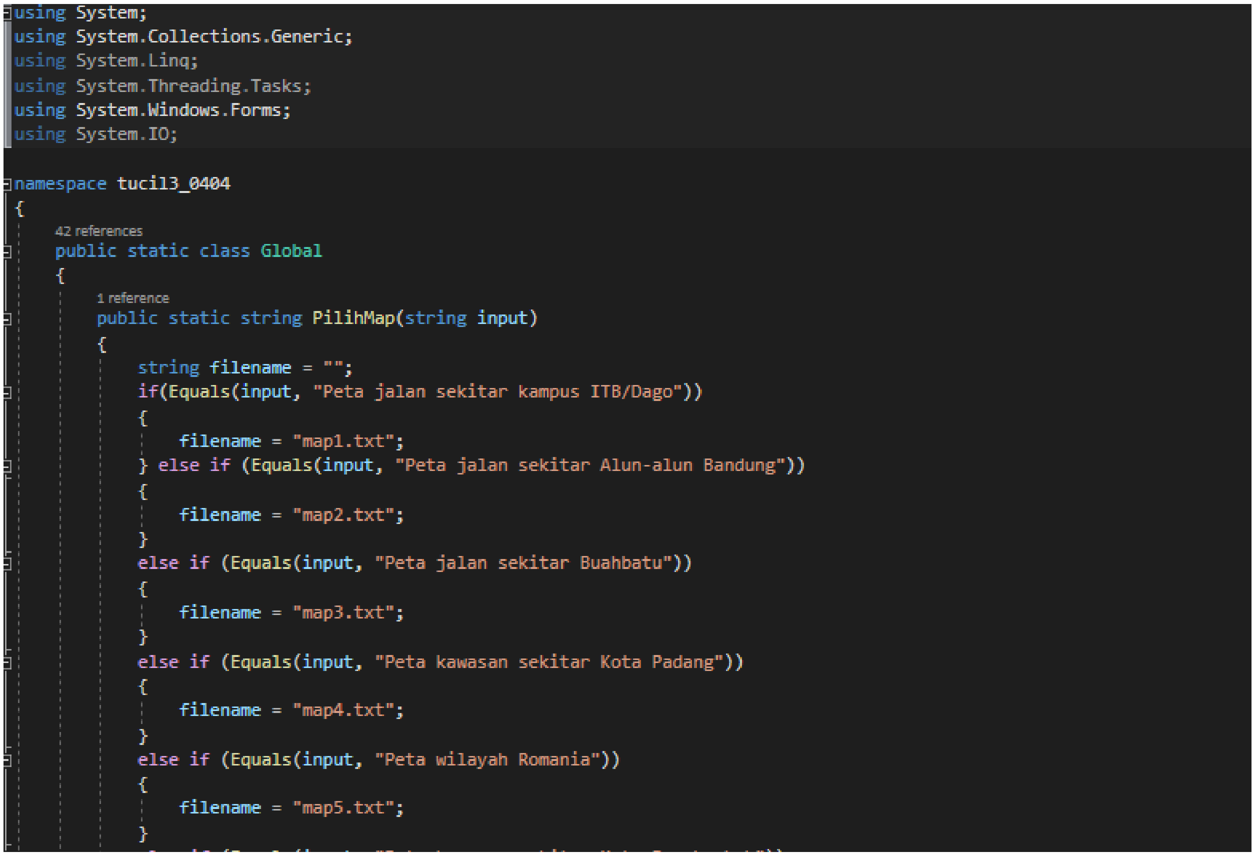


```
            // buat graf
            N = Global.JmlSimpul(path);
            Global.g = new graf(N);

            // buat graf MSAGL
            Global.viewer = new Microsoft.Msagl.GraphViewerGdi.GViewer();
            Global.graph = new Microsoft.Msagl.Drawing.Graph("graph");
            Global.nodes = new List<string>();

            //baca txt dan tambahkan ke msagl
            Global.g.CreateGraf(path);

            //panggil fungsi astar
            string msg = "";
            (List<string>, double) hasil = (new List<string>(), 0);
            astarsearch Astar = new astarsearch();
            try
            {
                hasil = Astar.astar(simpulasal, simpultujuan, Global.g);
                Global.g.AddMSAGLHasil(Global.graph, Global.nodes, hasil);
                foreach (string node in Global.nodes)
                {
                    Global.graph.FindNode(node).Attr.Color = Microsoft.Msagl.Drawing.Color.CadetBlue;
                    Global.graph.FindNode(node).Attr.FillColor = Microsoft.Msagl.Drawing.Color.CadetBlue;
                }
                foreach (string nama in hasil.Item1)
                {
                    //var temp = graph.AddEdge()
                    Global.graph.FindNode(nama).Attr.Color = Microsoft.Msagl.Drawing.Color.Coral;
                    Global.graph.FindNode(nama).Attr.FillColor = Microsoft.Msagl.Drawing.Color.Coral;
                }
                msg = "Jarak yang ditempuh: " + Convert.ToString(hasil.Item2);
            }
```

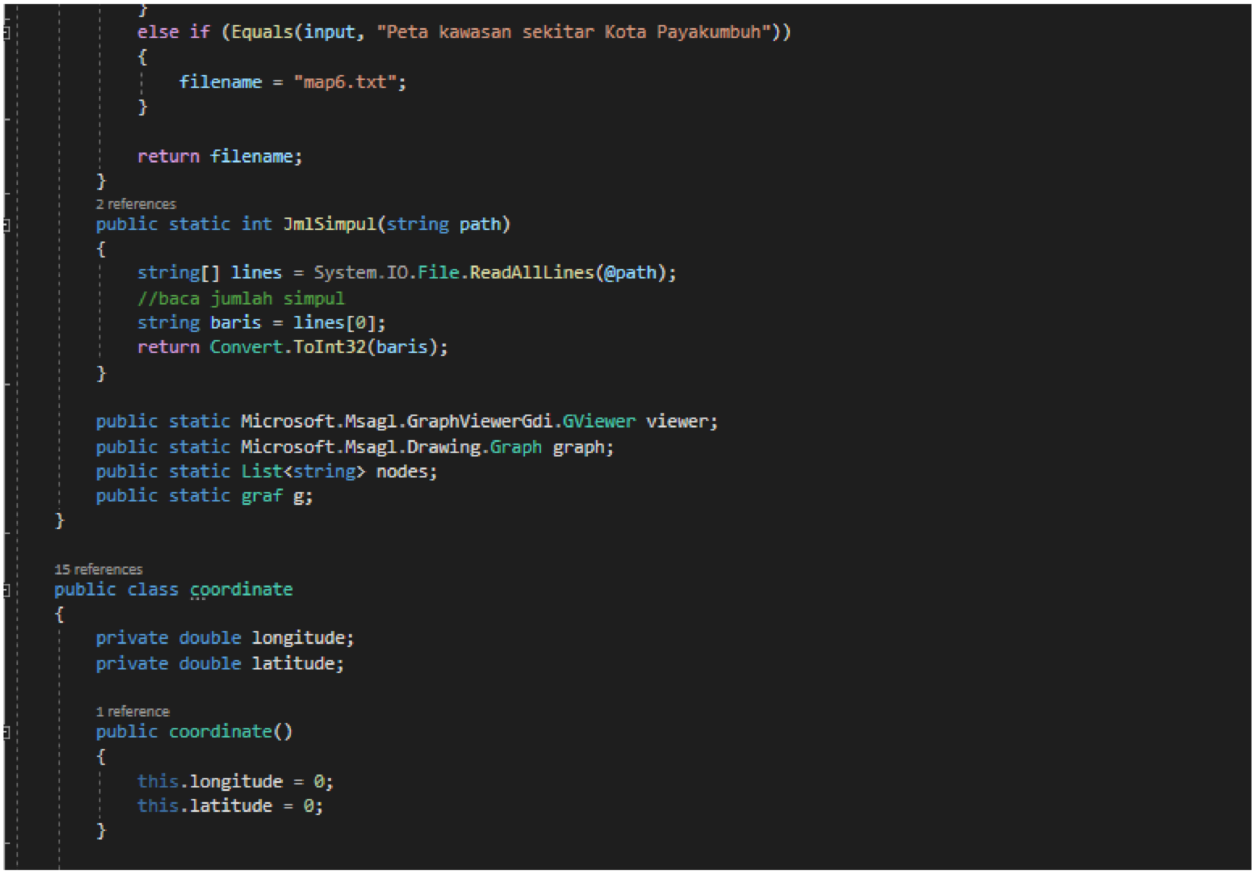


```
            catch (Exception)
            {
                msg = "Tidak ditemukan jalan :(";
            }

            //tampilkan hasil MSAGL sesuai hasil pencarian
            Global.viewer.Graph = Global.graph;
            Global.viewer.Dock = System.Windows.Forms.DockStyle.Fill;
            label5.Text = msg;
            groupBox1.Controls.Add(Global.viewer);
        }

        1 reference
        private void daftarMap_SelectedIndexChanged(object sender, EventArgs e)
        {
            mapterpilih = daftarMap.SelectedItem.ToString();
        }

        1 reference
        private void asal_SelectedIndexChanged(object sender, EventArgs e)
        {
            simpulasal = asal.SelectedItem.ToString();
        }

        1 reference
        private void tujuan_SelectedIndexChanged(object sender, EventArgs e)
        {
            simpultujuan = tujuan.SelectedItem.ToString();
        }
    }
}
```

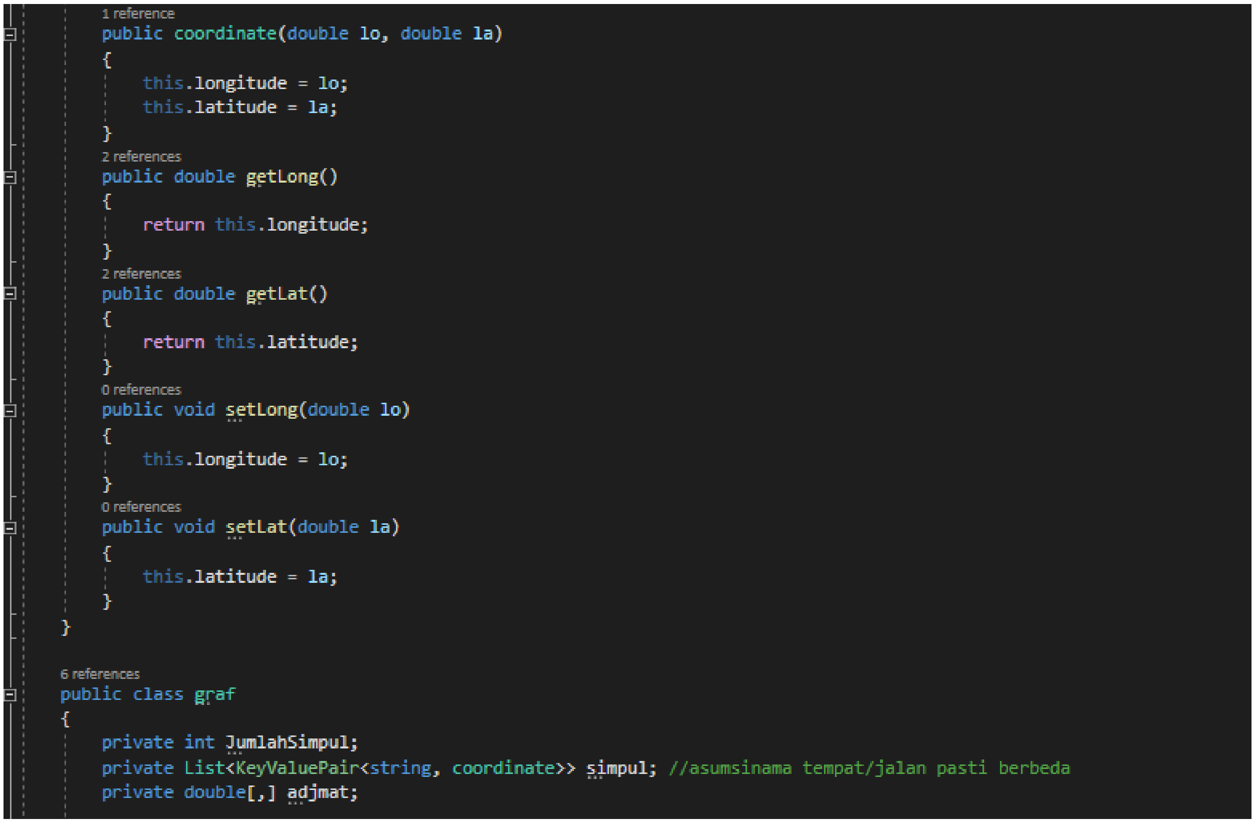
2. File Program.cs



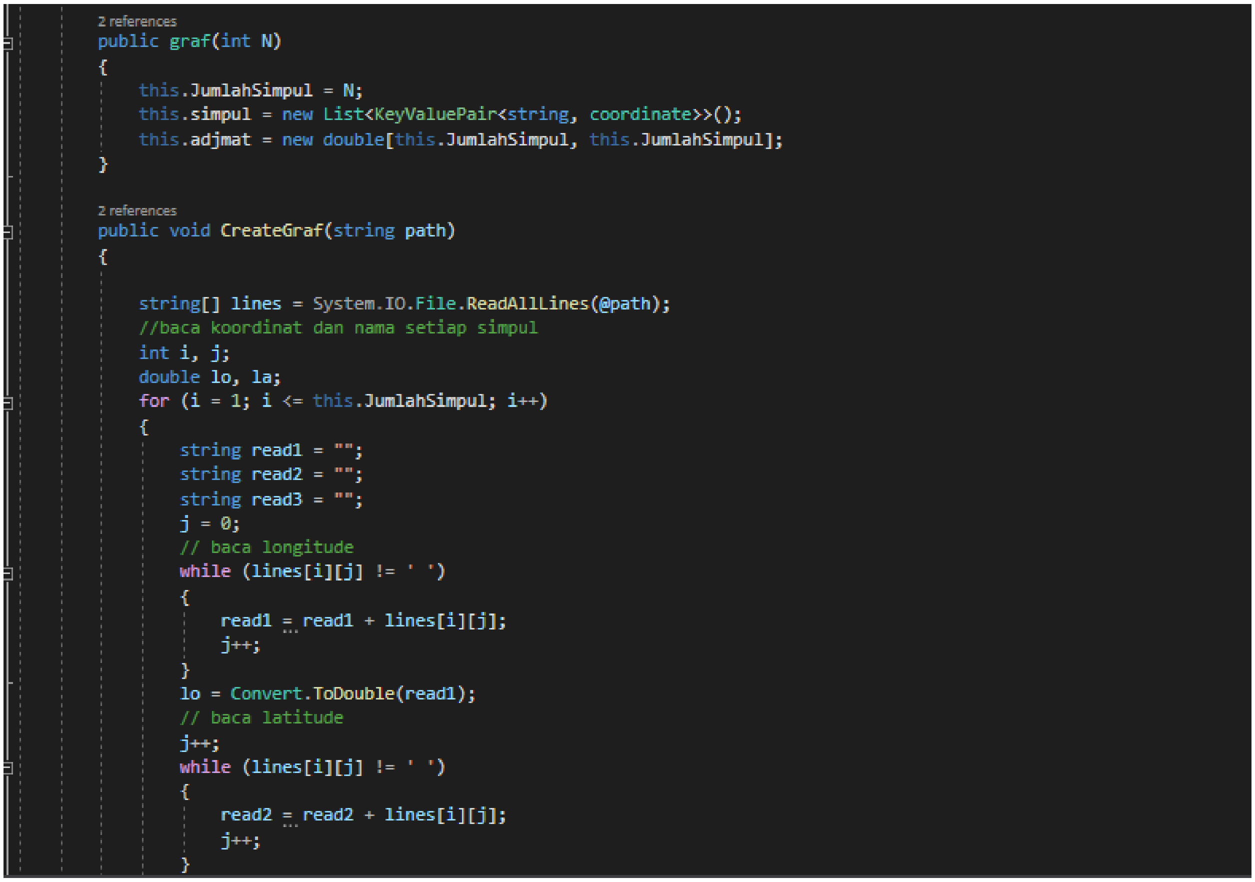
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.IO;

namespace tucil3_0404
{
    42 references
    public static class Global
    {
        1 reference
        public static string PilihMap(string input)
        {
            string filename = "";
            if(Equals(input, "Peta jalan sekitar kampus ITB/Dago"))
            {
                filename = "map1.txt";
            } else if (Equals(input, "Peta jalan sekitar Alun-alun Bandung"))
            {
                filename = "map2.txt";
            }
            else if (Equals(input, "Peta jalan sekitar Buahbatu"))
            {
                filename = "map3.txt";
            }
            else if (Equals(input, "Peta kawasan sekitar Kota Padang"))
            {
                filename = "map4.txt";
            }
            else if (Equals(input, "Peta wilayah Romania"))
            {
                filename = "map5.txt";
            }
```



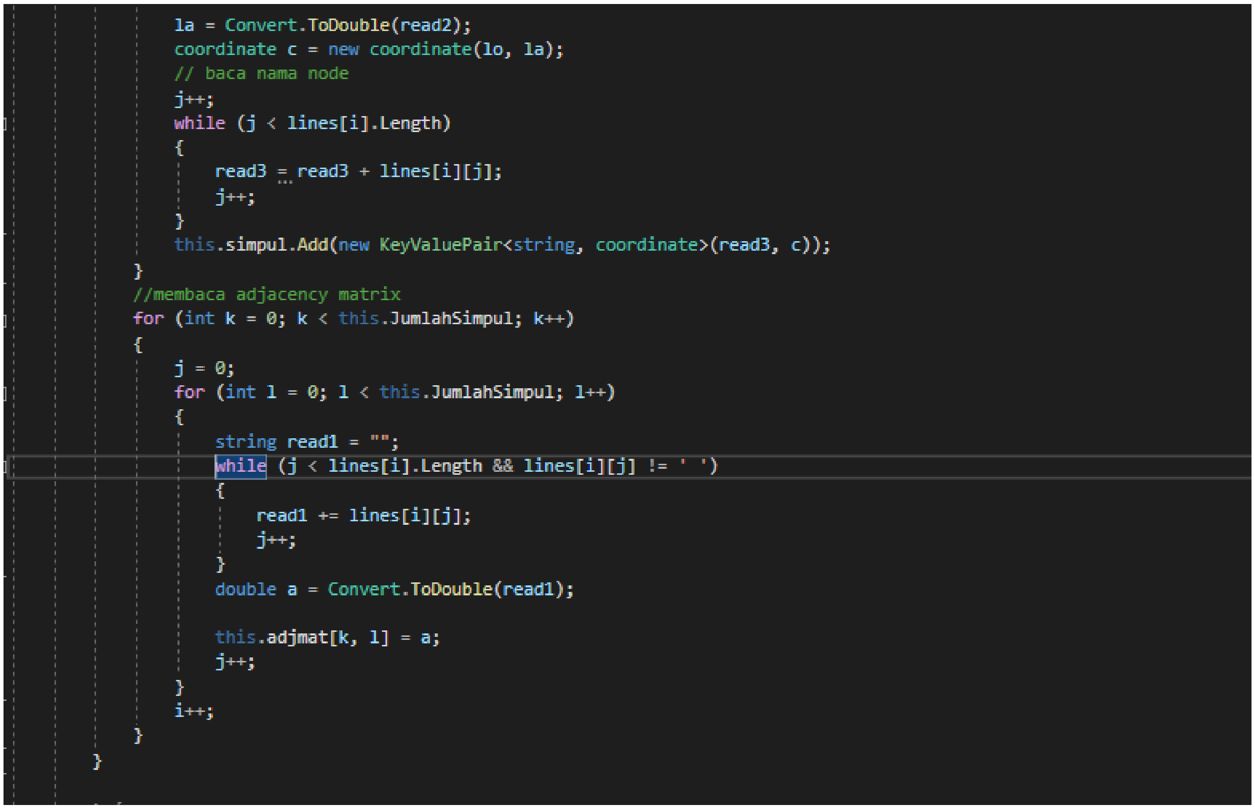
```
            }
            else if (Equals(input, "Peta kawasan sekitar Kota Payakumbuh"))
            {
                filename = "map6.txt";
            }

            return filename;
        }
        2 references
        public static int JmlSimpul(string path)
        {
            string[] lines = System.IO.File.ReadAllLines(@path);
            //baca jumlah simpul
            string baris = lines[0];
            return Convert.ToInt32(baris);
        }

        public static Microsoft.Msagl.GraphViewerGdi.GViewer viewer;
        public static Microsoft.Msagl.Drawing.Graph graph;
        public static List<string> nodes;
        public static graf g;
    }

    15 references
    public class coordinate
    {
        private double longitude;
        private double latitude;

        1 reference
        public coordinate()
        {
            this.longitude = 0;
            this.latitude = 0;
        }
```



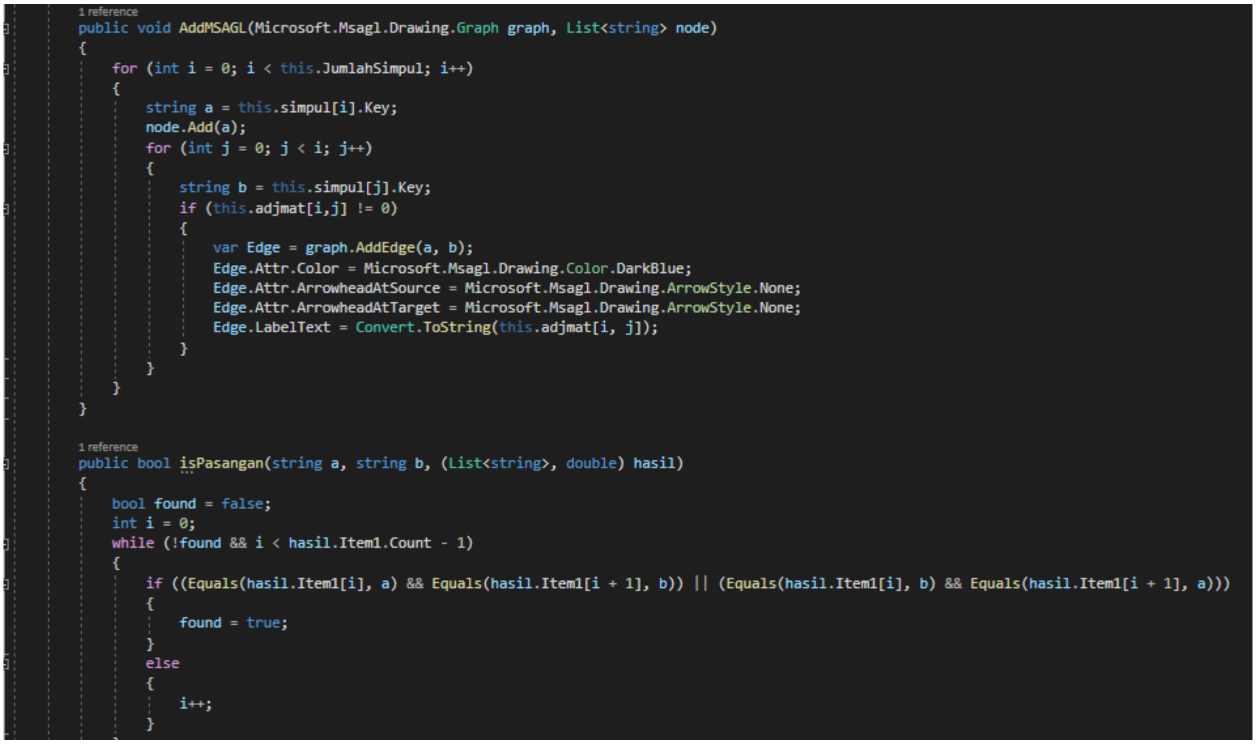
```
        1 reference
        public coordinate(double lo, double la)
        {
            this.longitude = lo;
            this.latitude = la;
        }
        2 references
        public double getLong()
        {
            return this.longitude;
        }
        2 references
        public double getLat()
        {
            return this.latitude;
        }
        0 references
        public void setLong(double lo)
        {
            this.longitude = lo;
        }
        0 references
        public void setLat(double la)
        {
            this.latitude = la;
        }
    }

    6 references
    public class graf
    {
        private int JumlahSimpul;
        private List<KeyValuePair<string, coordinate>> simpul; //asumsinama tempat/jalan pasti berbeda
        private double[,] adjmat;
```



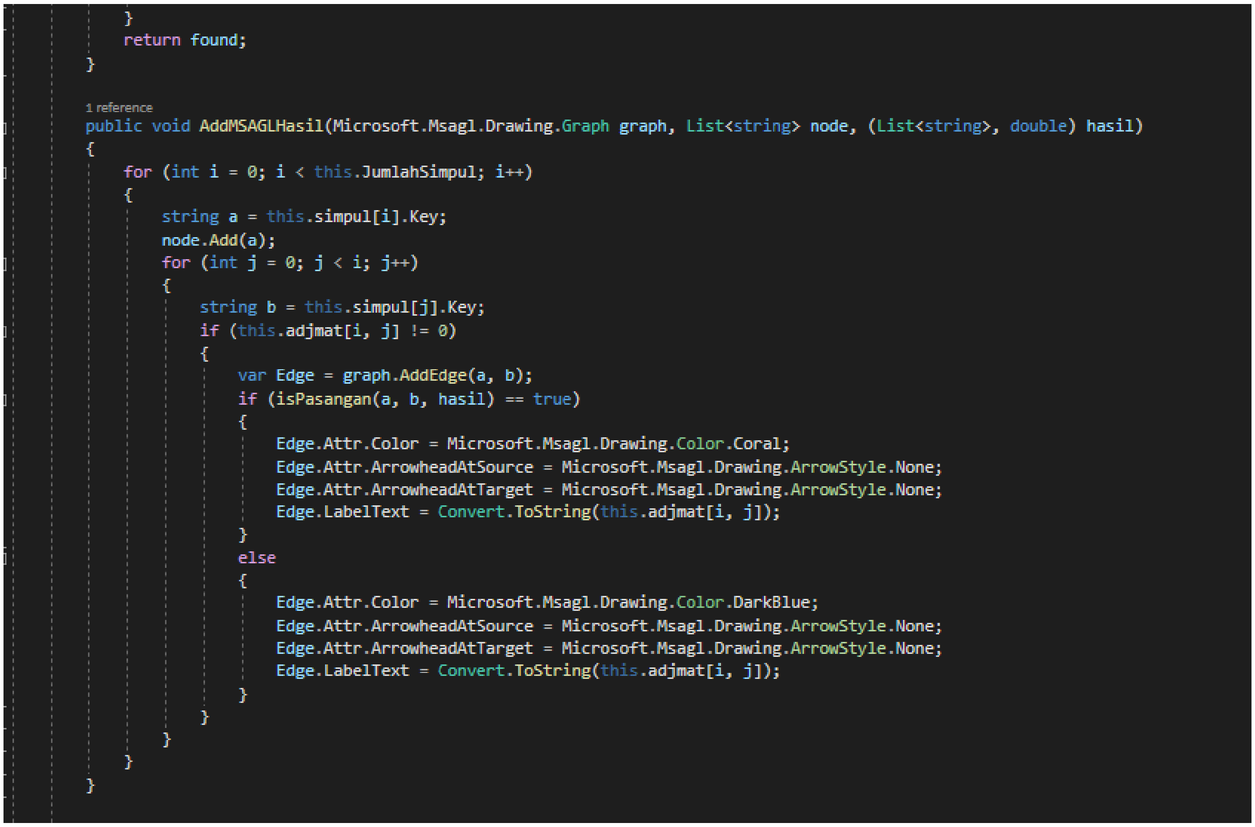
```
        2 references
        public graf(int N)
        {
            this.JumlahSimpul = N;
            this.simpul = new List<KeyValuePair<string, coordinate>>();
            this.adjmat = new double[this.JumlahSimpul, this.JumlahSimpul];
        }

        2 references
        public void CreateGraf(string path)
        {

            string[] lines = System.IO.File.ReadAllLines(@path);
            //baca koordinat dan nama setiap simpul
            int i, j;
            double lo, la;
            for (i = 1; i <= this.JumlahSimpul; i++)
            {
                string read1 = "";
                string read2 = "";
                string read3 = "";
                j = 0;
                // baca longitude
                while (lines[i][j] != ' ')
                {
                    read1 = read1 + lines[i][j];
                    j++;
                }
                lo = Convert.ToDouble(read1);
                // baca latitude
                j++;
                while (lines[i][j] != ' ')
                {
                    read2 = read2 + lines[i][j];
                    j++;
                }
```



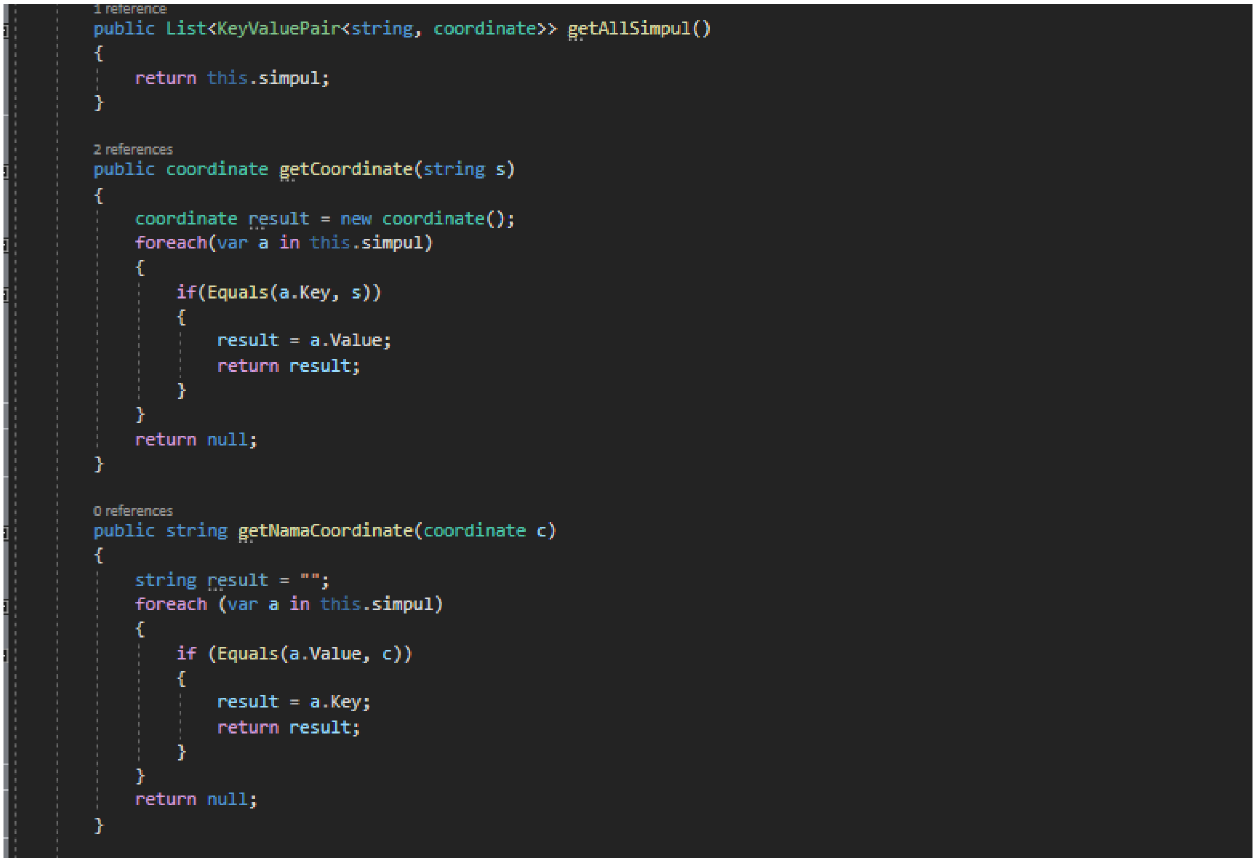
```
                la = Convert.ToDouble(read2);
                coordinate c = new coordinate(lo, la);
                // baca nama node
                j++;
                while (j < lines[i].Length)
                {
                    read3 = read3 + lines[i][j];
                    j++;
                }
                this.simpul.Add(new KeyValuePair<string, coordinate>(read3, c));
            }
            //membaca adjacency matrix
            for (int k = 0; k < this.JumlahSimpul; k++)
            {
                j = 0;
                for (int l = 0; l < this.JumlahSimpul; l++)
                {
                    string read1 = "";
                    while (j < lines[i].Length && lines[i][j] != ' ')
                    {
                        read1 += lines[i][j];
                        j++;
                    }
                    double a = Convert.ToDouble(read1);

                    this.adjmat[k, l] = a;
                    j++;
                }
                i++;
            }
        }
```



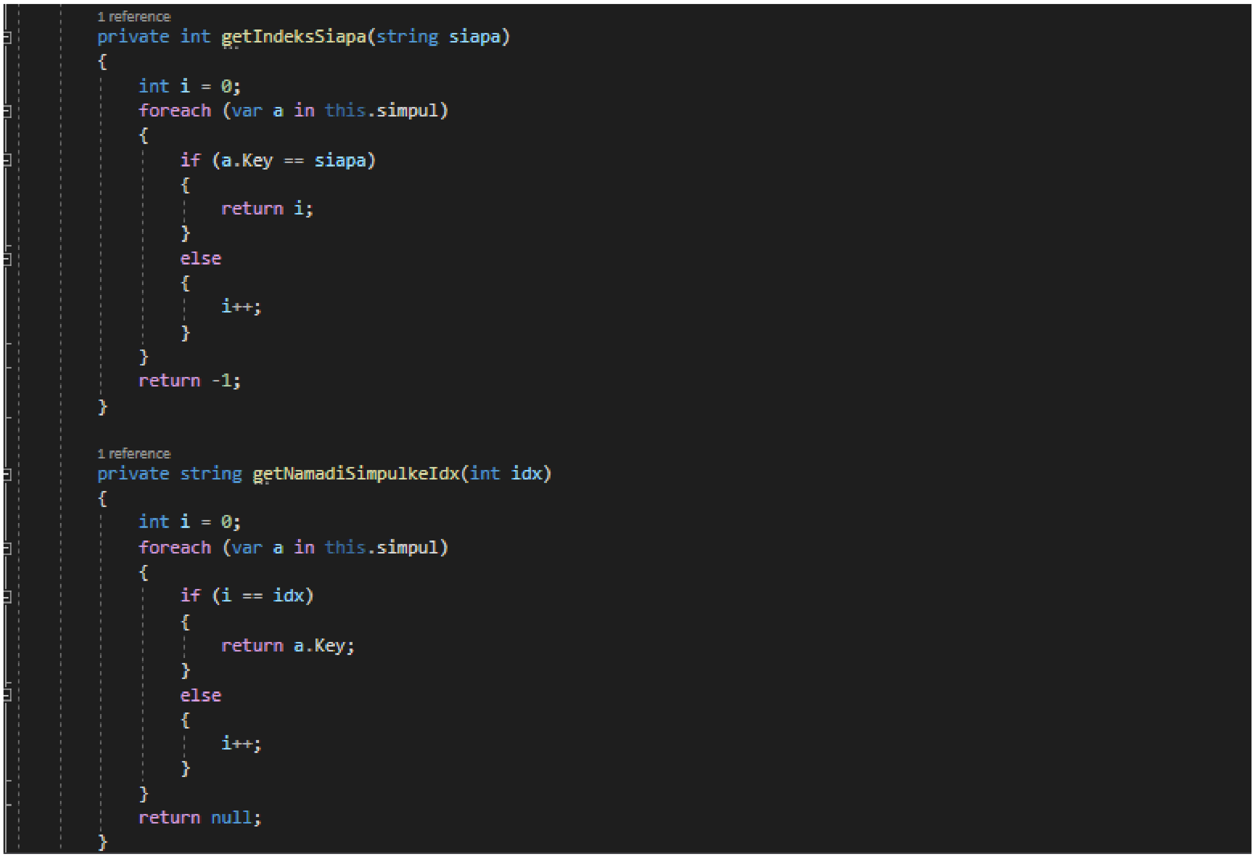
```
1 reference
public void AddMSAGL(Microsoft.Msagl.Drawing.Graph graph, List<string> node)
{
    for (int i = 0; i < this.JumlahSimpul; i++)
    {
        string a = this.simpul[i].Key;
        node.Add(a);
        for (int j = 0; j < i; j++)
        {
            string b = this.simpul[j].Key;
            if (this.adjmat[i,j] != 0)
            {
                var Edge = graph.AddEdge(a, b);
                Edge.Attr.Color = Microsoft.Msagl.Drawing.Color.DarkBlue;
                Edge.Attr.ArrowheadAtSource = Microsoft.Msagl.Drawing.ArrowStyle.None;
                Edge.Attr.ArrowheadAtTarget = Microsoft.Msagl.Drawing.ArrowStyle.None;
                Edge.LabelText = Convert.ToString(this.adjmat[i, j]);
            }
        }
    }
}

1 reference
public bool isPasangan(string a, string b, (List<string>, double) hasil)
{
    bool found = false;
    int i = 0;
    while (!found && i < hasil.Item1.Count - 1)
    {
        if ((Equals(hasil.Item1[i], a) && Equals(hasil.Item1[i + 1], b)) || (Equals(hasil.Item1[i], b) && Equals(hasil.Item1[i + 1], a)))
        {
            found = true;
        }
        else
        {
            i++;
        }
```



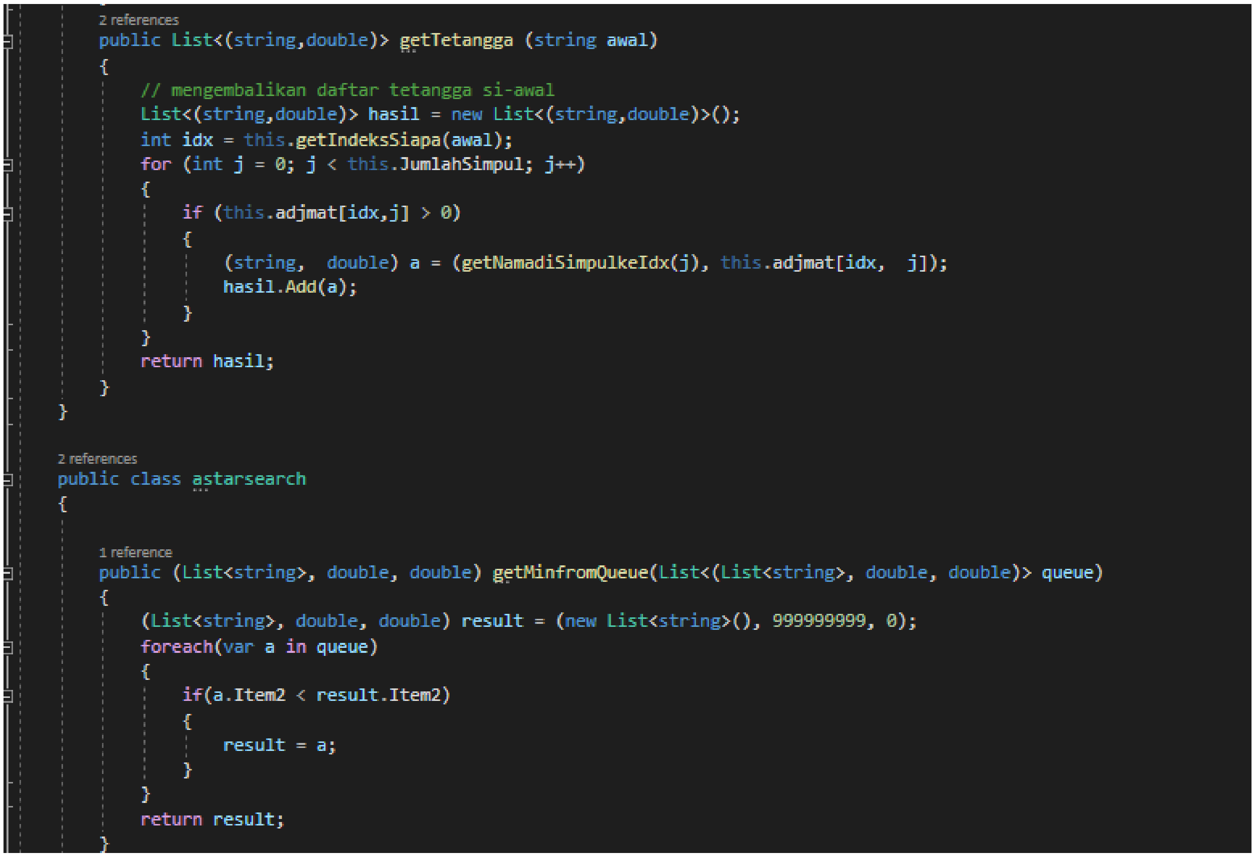
```
    }
    return found;
}

1 reference
public void AddMSAGLHasil(Microsoft.Msagl.Drawing.Graph graph, List<string> node, (List<string>, double) hasil)
{
    for (int i = 0; i < this.JumlahSimpul; i++)
    {
        string a = this.simpul[i].Key;
        node.Add(a);
        for (int j = 0; j < i; j++)
        {
            string b = this.simpul[j].Key;
            if (this.adjmat[i, j] != 0)
            {
                var Edge = graph.AddEdge(a, b);
                if (isPasangan(a, b, hasil) == true)
                {
                    Edge.Attr.Color = Microsoft.Msagl.Drawing.Color.Coral;
                    Edge.Attr.ArrowheadAtSource = Microsoft.Msagl.Drawing.ArrowStyle.None;
                    Edge.Attr.ArrowheadAtTarget = Microsoft.Msagl.Drawing.ArrowStyle.None;
                    Edge.LabelText = Convert.ToString(this.adjmat[i, j]);
                }
                else
                {
                    Edge.Attr.Color = Microsoft.Msagl.Drawing.Color.DarkBlue;
                    Edge.Attr.ArrowheadAtSource = Microsoft.Msagl.Drawing.ArrowStyle.None;
                    Edge.Attr.ArrowheadAtTarget = Microsoft.Msagl.Drawing.ArrowStyle.None;
                    Edge.LabelText = Convert.ToString(this.adjmat[i, j]);
                }
            }
        }
    }
}
```



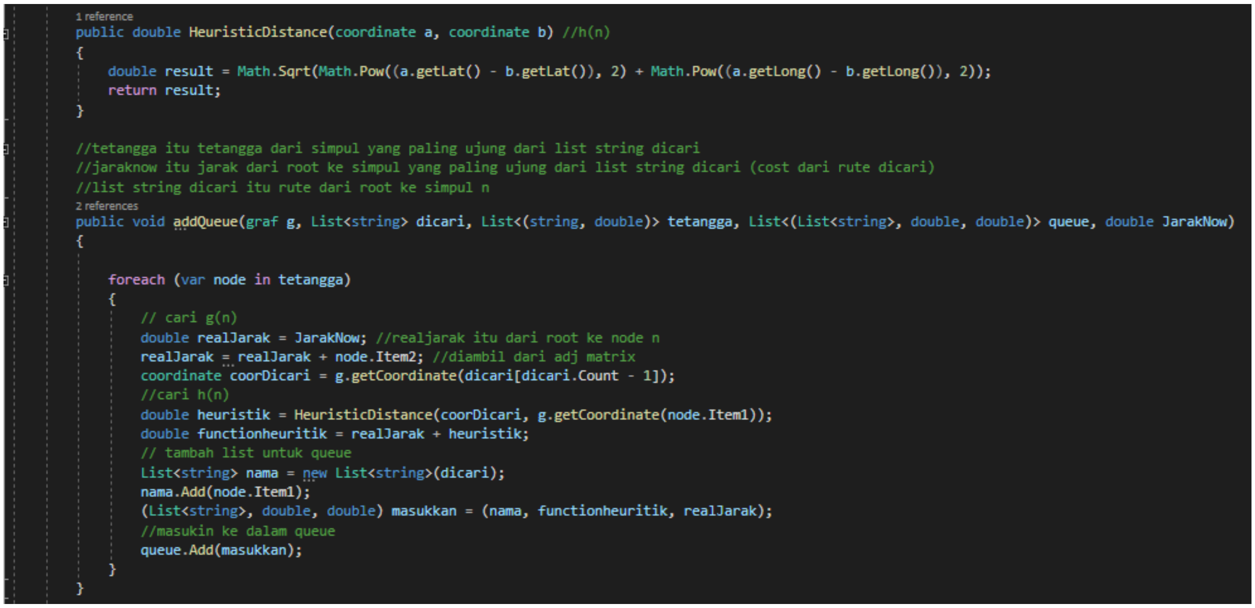
```
1 reference
public List<KeyValuePair<string, coordinate>> getAllSimpul()
{
    return this.simpul;
}

2 references
public coordinate getCoordinate(string s)
{
    coordinate result = new coordinate();
    foreach(var a in this.simpul)
    {
        if(Equals(a.Key, s))
        {
            result = a.Value;
            return result;
        }
    }
    return null;
}

0 references
public string getNamaCoordinate(coordinate c)
{
    string result = "";
    foreach (var a in this.simpul)
    {
        if (Equals(a.Value, c))
        {
            result = a.Key;
            return result;
        }
    }
    return null;
}
```



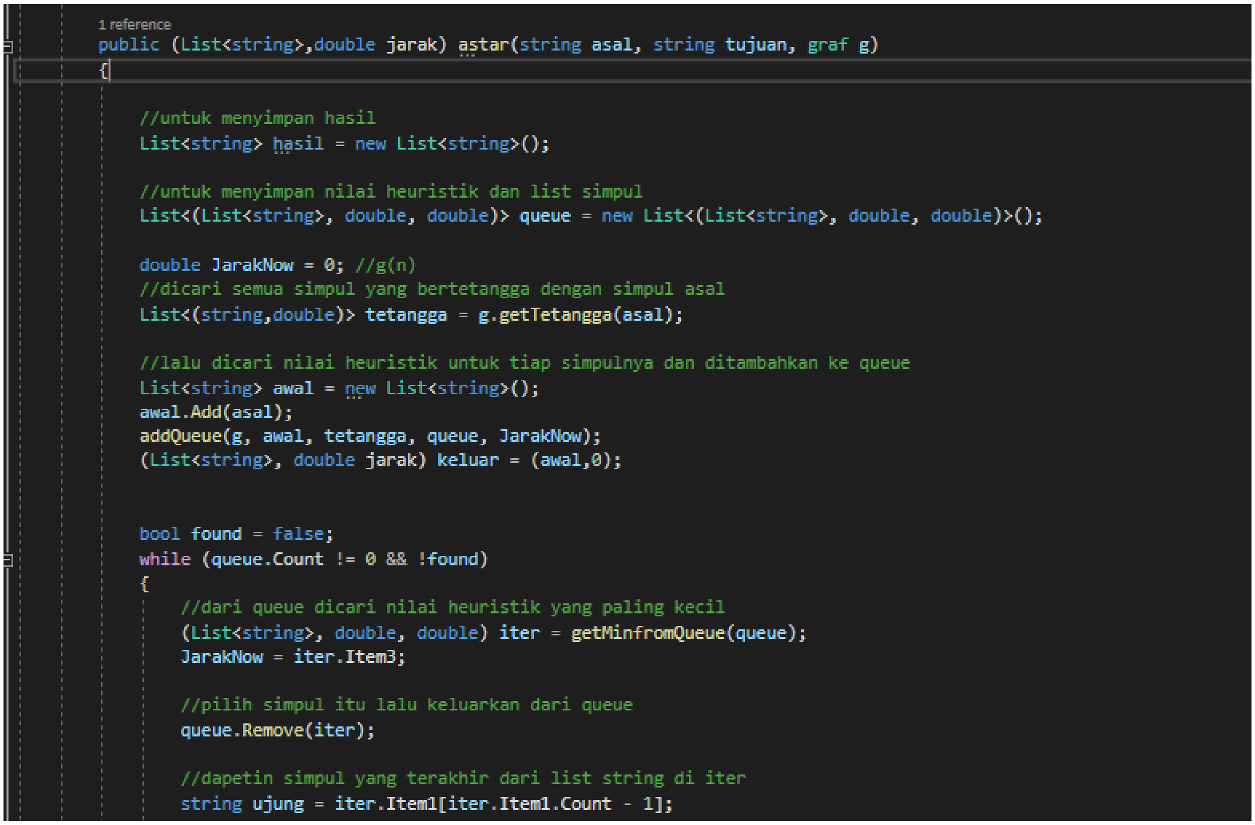
```
        1 reference
        private int getIndeksSiapa(string siapa)
        {
            int i = 0;
            foreach (var a in this.simpul)
            {
                if (a.Key == siapa)
                {
                    return i;
                }
                else
                {
                    i++;
                }
            }
            return -1;
        }

        1 reference
        private string getNamadiSimpulkeIdx(int idx)
        {
            int i = 0;
            foreach (var a in this.simpul)
            {
                if (i == idx)
                {
                    return a.Key;
                }
                else
                {
                    i++;
                }
            }
            return null;
        }
```



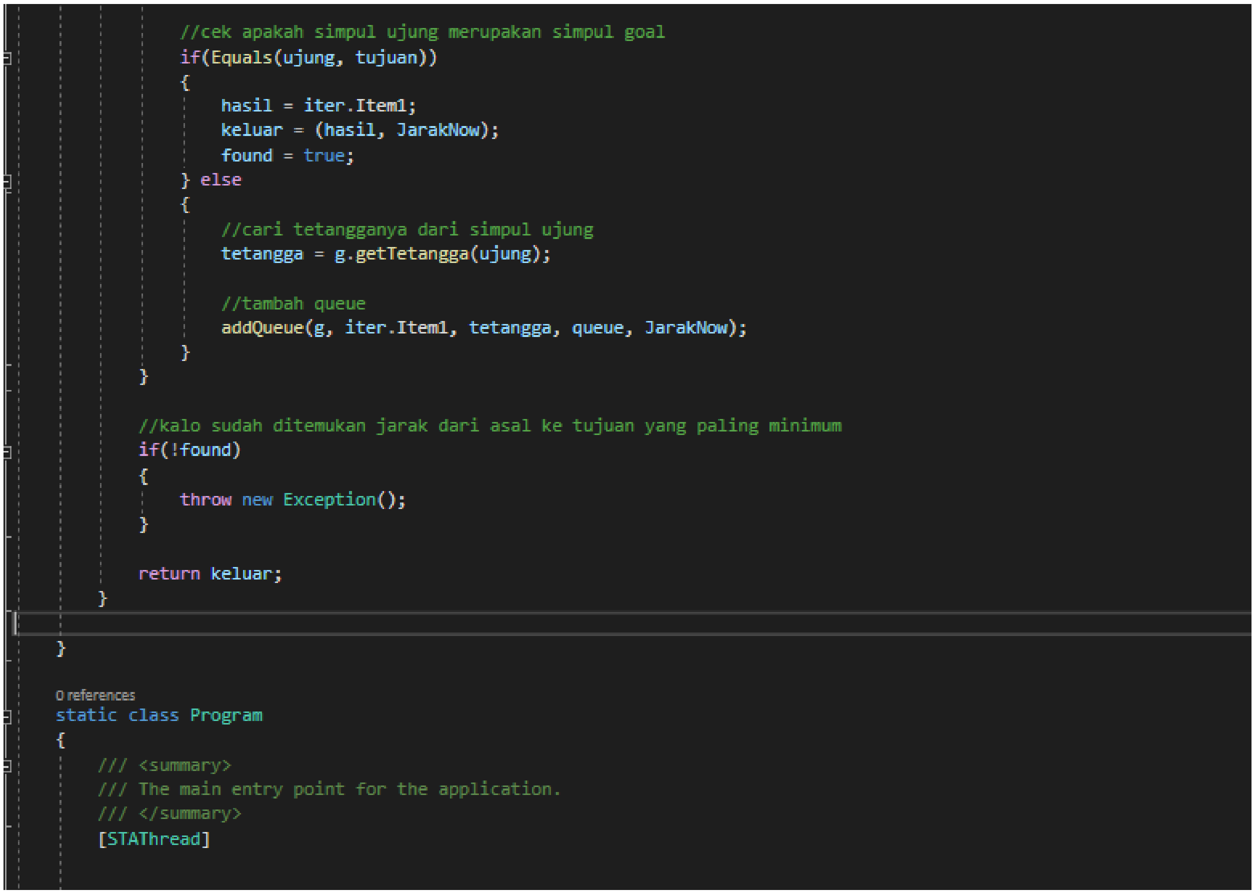
```
        2 references
        public List<(string,double)> getTetangga (string awal)
        {
            // mengembalikan daftar tetangga si-awal
            List<(string,double)> hasil = new List<(string,double)>();
            int idx = this.getIndeksSiapa(awal);
            for (int j = 0; j < this.JumlahSimpul; j++)
            {
                if (this.adjmat[idx,j] > 0)
                {
                    (string,  double) a = (getNamadiSimpulkeIdx(j), this.adjmat[idx,  j]);
                    hasil.Add(a);
                }
            }
            return hasil;
        }
    }

    2 references
    public class astarsearch
    {

        1 reference
        public (List<string>, double, double) getMinfromQueue(List<(List<string>, double, double)> queue)
        {
            (List<string>, double, double) result = (new List<string>(), 999999999, 0);
            foreach(var a in queue)
            {
                if(a.Item2 < result.Item2)
                {
                    result = a;
                }
            }
            return result;
        }
```



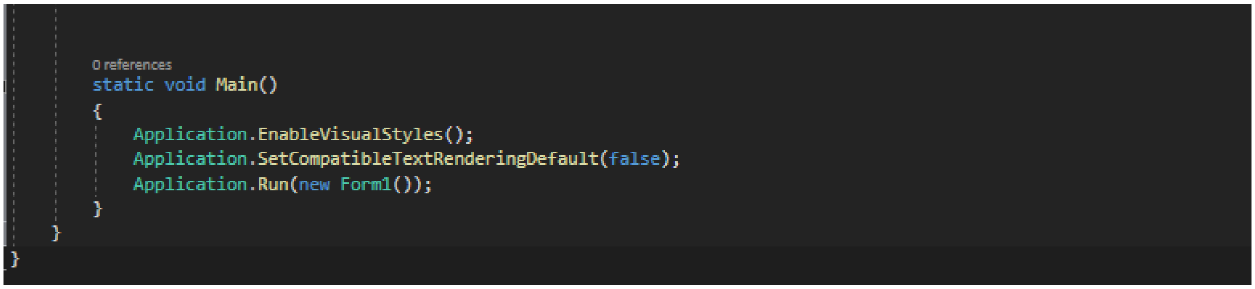
```
1 reference
public double HeuristicDistance(coordinate a, coordinate b) //h(n)
{
    double result = Math.Sqrt(Math.Pow((a.getLat() - b.getLat()), 2) + Math.Pow((a.getLong() - b.getLong()), 2));
    return result;
}

//tetangga itu tetangga dari simpul yang paling ujung dari list string dicari
//jaraknow itu jarak dari root ke simpul yang paling ujung dari list string dicari (cost dari rute dicari)
//list string dicari itu rute dari root ke simpul n
2 references
public void addQueue(graf g, List<string> dicari, List<(string, double)> tetangga, List<(List<string>, double, double)> queue, double JarakNow)
{

    foreach (var node in tetangga)
    {
        // cari g(n)
        double realJarak = JarakNow; //realjarak itu dari root ke node n
        realJarak = realJarak + node.Item2; //diambil dari adj matrix
        coordinate coorDicari = g.getCoordinate(dicari[dicari.Count - 1]);
        //cari h(n)
        double heuristik = HeuristicDistance(coorDicari, g.getCoordinate(node.Item1));
        double functionheuritik = realJarak + heuristik;
        // tambah list untuk queue
        List<string> nama = new List<string>(dicari);
        nama.Add(node.Item1);
        (List<string>, double, double) masukkan = (nama, functionheuritik, realJarak);
        //masukin ke dalam queue
        queue.Add(masukkan);
    }
}
```



```
1 reference
public (List<string>,double jarak) astar(string asal, string tujuan, graf g)
{

    //untuk menyimpan hasil
    List<string> hasil = new List<string>();

    //untuk menyimpan nilai heuristik dan list simpul
    List<(List<string>, double, double)> queue = new List<(List<string>, double, double)>();

    double JarakNow = 0; //g(n)
    //dicari semua simpul yang bertetangga dengan simpul asal
    List<(string,double)> tetangga = g.getTetangga(asal);

    //lalu dicari nilai heuristik untuk tiap simpulnya dan ditambahkan ke queue
    List<string> awal = new List<string>();
    awal.Add(asal);
    addQueue(g, awal, tetangga, queue, JarakNow);
    (List<string>, double jarak) keluar = (awal,0);


    bool found = false;
    while (queue.Count != 0 && !found)
    {
        //dari queue dicari nilai heuristik yang paling kecil
        (List<string>, double, double) iter = getMinfromQueue(queue);
        JarakNow = iter.Item3;

        //pilih simpul itu lalu keluarkan dari queue
        queue.Remove(iter);

        //dapetin simpul yang terakhir dari list string di iter
        string ujung = iter.Item1[iter.Item1.Count - 1];
```



```
                //cek apakah simpul ujung merupakan simpul goal
                if(Equals(ujung, tujuan))
                {
                    hasil = iter.Item1;
                    keluar = (hasil, JarakNow);
                    found = true;
                } else
                {
                    //cari tetangganya dari simpul ujung
                    tetangga = g.getTetangga(ujung);

                    //tambah queue
                    addQueue(g, iter.Item1, tetangga, queue, JarakNow);
                }
            }

            //kalo sudah ditemukan jarak dari asal ke tujuan yang paling minimum
            if(!found)
            {
                throw new Exception();
            }

            return keluar;
        }

    }

    0 references
    static class Program
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
```



```


        0 references
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new Form1());
        }
    }
}
```

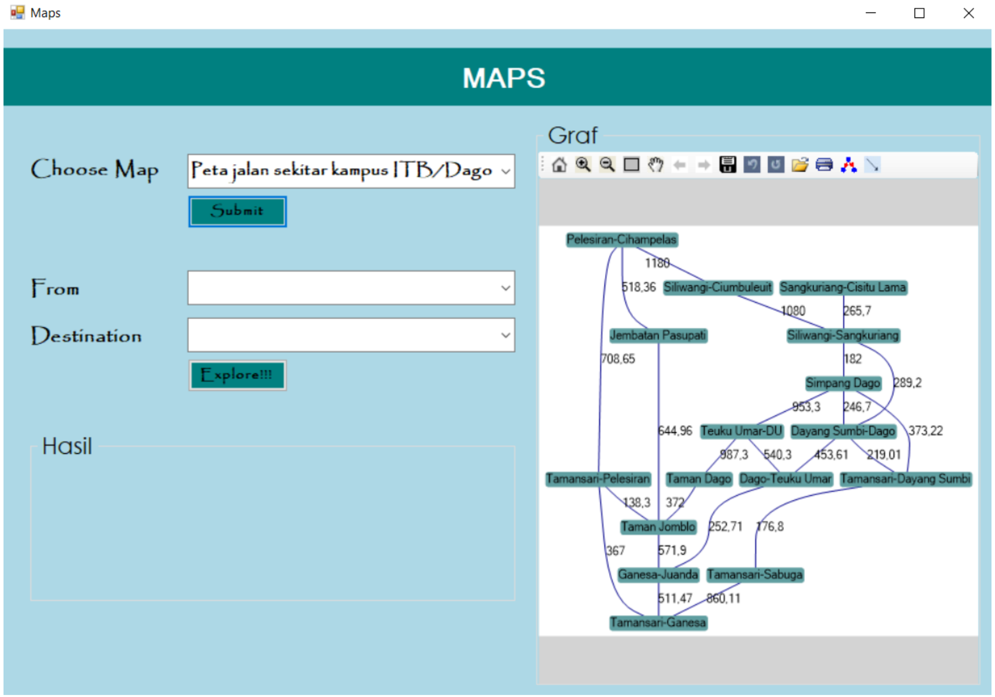
B. Graf Input

1. Peta jalan sekitar kampus ITB/Dago
    - Input file txt

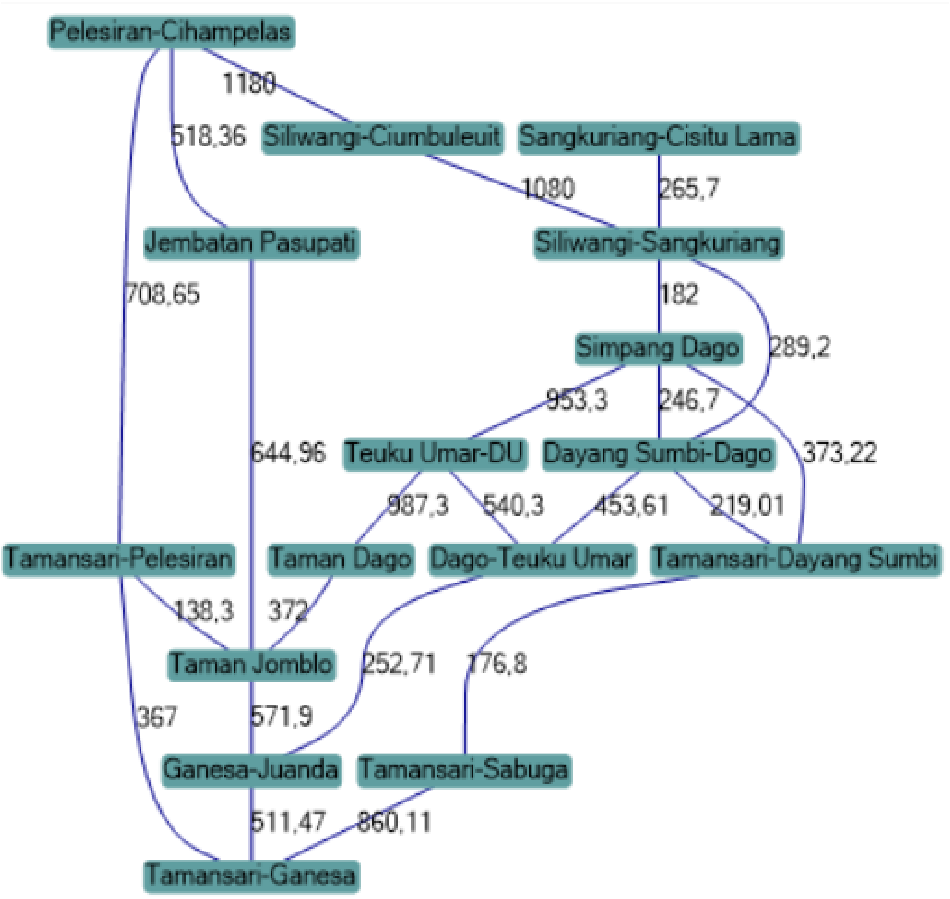
```
16
-6,893848651218295 107,60846317380707 Tamansari-Ganesa
-6,893748721200304 107,61287996656958 Ganesa-Juanda
-6,8914906463422385 107,61315891632162 Dago-Teuku Umar
-6,887658726990241 107,60993219853344 Tamansari-Sabuga
-6,887083273817282 107,61148059597167 Tamansari-Dayang Sumbi
-6,887396145411665 107,61344578017354 Dayang Sumbi-Dago
-6,898098574710648 107,60950852644122 Taman Jomblo
-6,898981103552023 107,6127128222436 Taman Dago
-6,892400522068246 107,61780140087448 Teuku Umar-DU
```

```
-6,885217845776812 107,61372207739318 Simpang Dago
-6,884903586895761 107,61197376414513 Siliwangi-Sangkuriang
-6,883703356593794 107,61122918933259 Sangkuriang-Cisitu Lama
-6,883323192659634 107,6049210494182 Siliwangi-Ciumbuleuit
-6,90022991951274 107,60418047440548 Jembatan Pasupati
-6,89684588262392 107,60962329924541 Tamansari-Pelesiran
-6,895592752274568 107,6039473149935 Pelesiran-Cihampelas
0 511,47 0 860,11 0 0 0 0 0 0 0 0 0 0 367 0
511,47 0 252,71 0 0 0 0 571,9 0 0 0 0 0 0 0 0
0 252,71 0 0 0 453,61 0 0 540,3 0 0 0 0 0 0 0
860,11 0 0 0 176,80 0 0 0 0 0 0 0 0 0 0 0
0 0 0 176,80 0 219,01 0 0 0 373,22 289,2 0 0 0 0 0
0 0 453,61 0 219,01 0 0 0 0 246,7 0 0 0 0 0 0
0 571,9 0 0 0 0 0 372 0 0 0 0 0 644,95 138,3 0
0 0 0 0 0 0 372 0 987,3 0 0 0 0 0 0 0
0 0 540,3 0 0 0 0 987,3 0 953,3 0 0 0 0 0 0
0 0 0 0 373,22 246,7 0 0 953,3 0 182 0 0 0 0 0
0 0 0 0 0 289,2 0 0 0 182 0 265,7 1080 0 0 0
0 0 0 0 0 0 0 0 0 0 265,7 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 1080 0 0 0 0 1180
0 0 0 0 0 0 644,96 0 0 0 0 0 0 0 0 518,36
367 0 0 0 0 0 138,3 0 0 0 0 0 0 0 0 708,65
0 0 0 0 0 0 0 0 0 0 0 0 1180 518,36 708,65 0
```

- Visualisasi Graf



Choose Map: Peta jalan sekitar kampus ITB/Dago

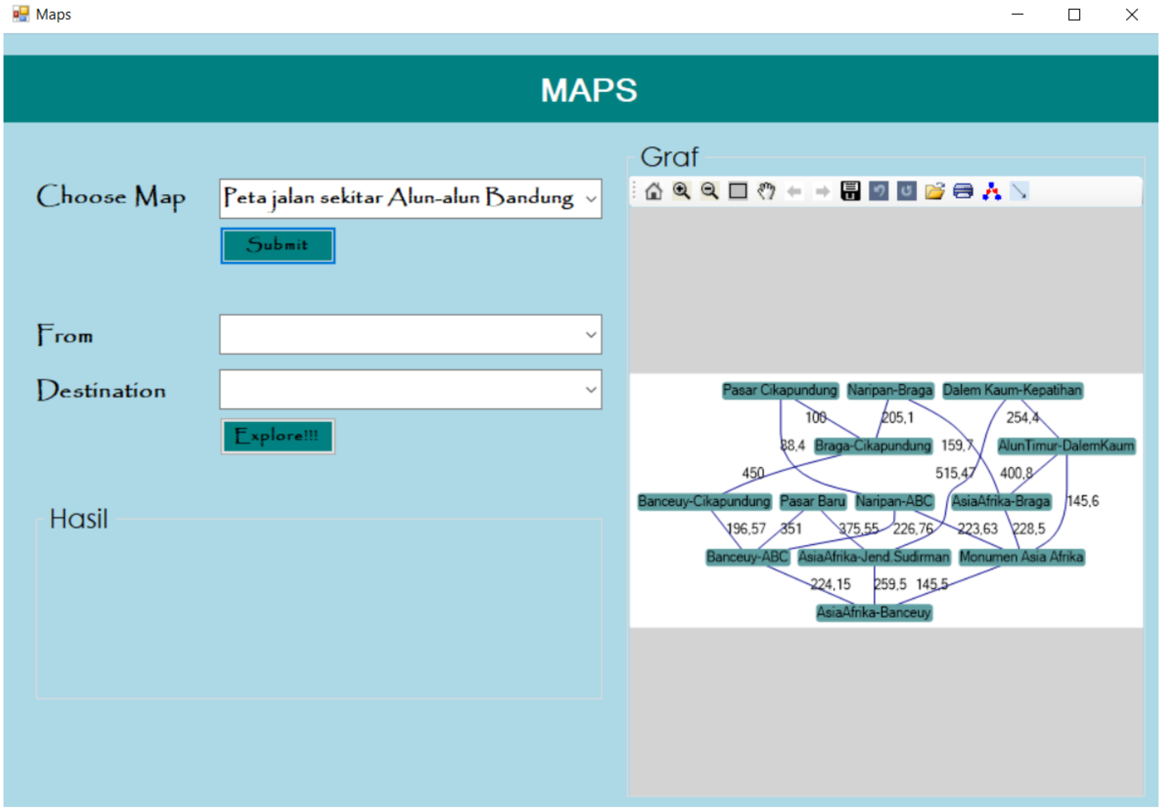


2. Peta jalan sekitar Alun-alun Bandung
    - Input file txt

```
13
-6,921043203272078 107,60646421076471 AsiaAfrika-Banceuy
-6,920780852599953 107,60408101163092 AsiaAfrika-Jend.Sudirman
-6,921285938866149 107,60768562556657 Monumen Asia Afrika
-6,921434828671815 107,60979843105106 AsiaAfrika-Braga
-6,922580475886105 107,60761370584542 AlunTimur-DalemKaum
-6,91886784442933 107,60667949763908 Banceuy-ABC
-6,918205352030106 107,60651952651492 Banceuy-Cikapundung
-6,917463786998024 107,60432408109426 Pasar Baru
-6,917908034658428 107,6094070567096 Braga-Cikapundung
-6,919698725559906 107,60991031465896 Naripan-Braga
-6,923434887775142 107,6062656546735 Dalem Kaum-Kepatihan
-6,9195596526168455 107,60838601823282 Naripan-ABC
-6,919065063477976 107,60848557962171 Pasar Cikapundung
0 259,5 145,5 0 0 224,15 0 0 0 0 0 0 0
259,5 0 0 0 0 0 0 375,55 0 0 515,47 0 0
145,5 0 0 228,5 145,6 0 0 0 0 0 0 223,63 0
0 0 228,5 0 400,8 0 0 0 0 159,7 0 0 0
```

```
0 0 145,6 400,8 0 0 0 0 0 0 254,4 0 0
224,15 0 0 0 0 0 196,57 351 0 0 0 226,76 0
0 0 0 0 0 196,57 0 0 450 0 0 0 0
0 375,55 0 0 0 351 0 0 0 0 0 0 0
0 0 0 0 0 0 450 0 0 205,1 0 0 100
0 0 0 159,7 0 0 0 0 205,1 0 0 0 0
0 515,47 0 0 254,4 0 0 0 0 0 0 0 0
0 0 223,63 0 0 226,76 0 0 0 0 0 0 88,4
0 0 0 0 0 0 0 0 100 0 0 88,4 0
```

- Visualisasi Graf



Maps

MAPS

Choose Map: Peta jalan sekitar Alun-alun Bandung

Submit

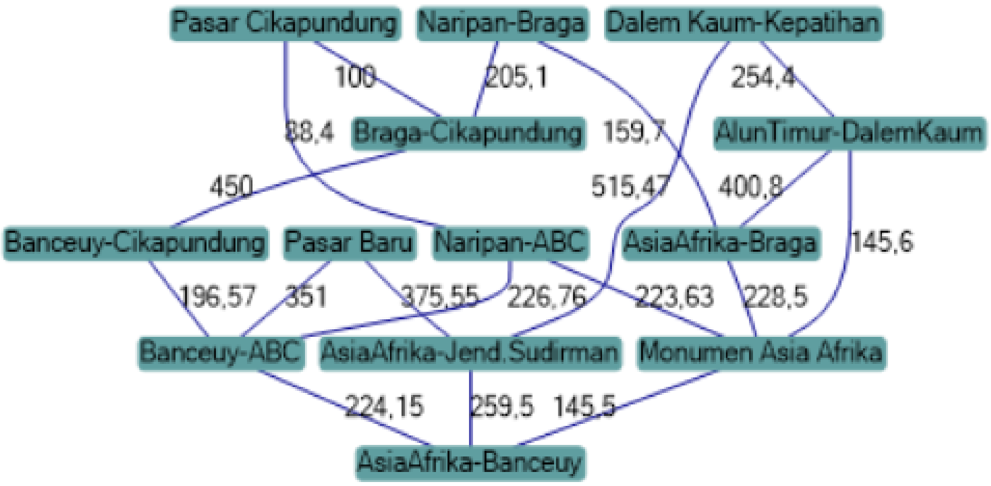
From

Destination

Explore!!!

Hasil

Graf

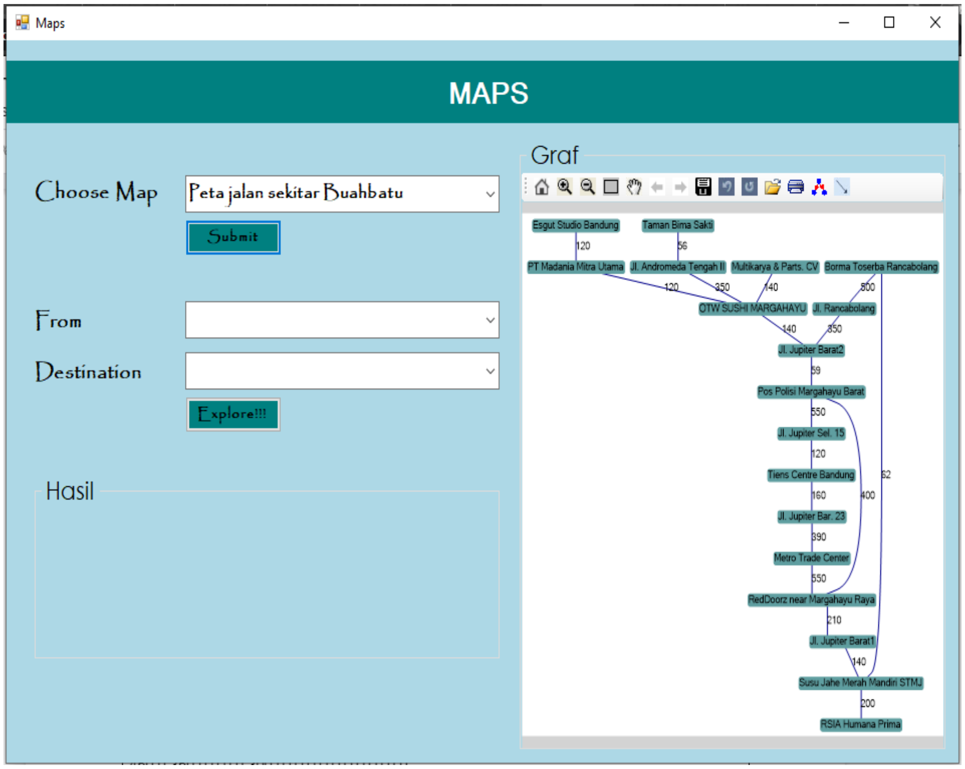


3. Peta jalan sekitar Buahbatu
    - Input file txt

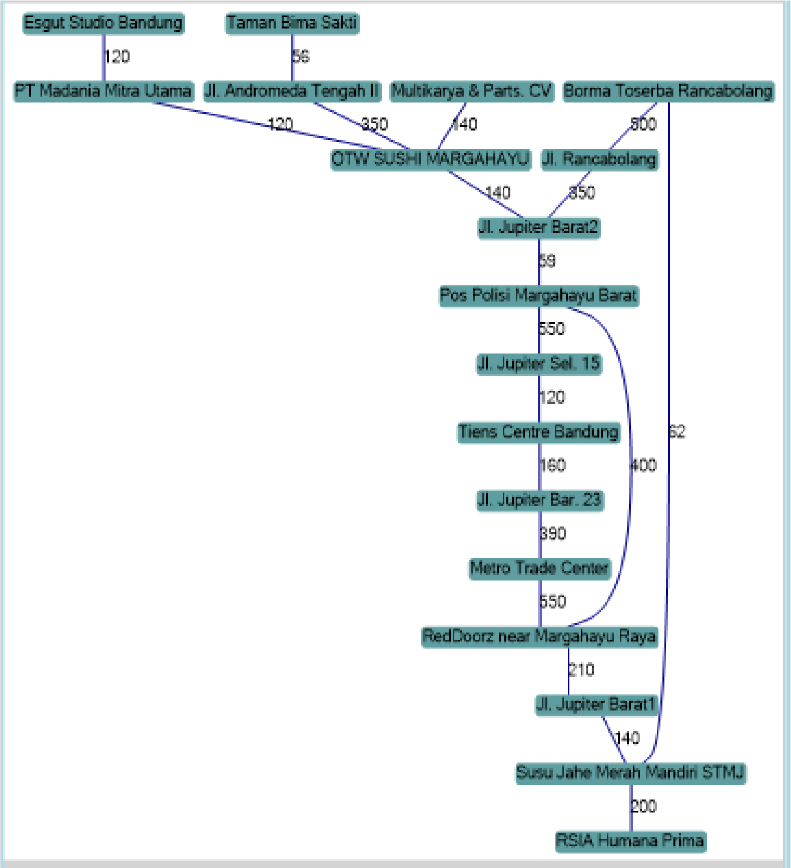
```
18
-6,9404746935582855 107,66403502414212 RSIA Humana Prima
-6,942696 107,663619 Susu Jahe Merah Mandiri STMJ
-6,942248 107,662491 Jl. Jupiter Barat1
-6,943564429141604 107,66145364064346 RedDoorz near Margahayu Raya
-6,943197246494815 107,658954591187 Metro Trade Center
-6,944419 107,657572 Jl. Jupiter Bar. 23
-6,944273083695128 107,65613014808076 Tiens Centre Bandung
-6,945264 107,656175 Jl. Jupiter Sel. 15
-6,946710703115815 107,65953353066202 Pos Polisi Margahayu Barat
-6,947263 107,659376 Jl. Jupiter Barat2
-6,948113887994259 107,65868393585028 OTW SUSHI MARGAHAYU
-6,948558464013932 107,65963226846486 Multikarya & Parts. CV
-6,948145467210398 107,65760220691097 PT Madania Mitra Utama
-6,947719842124904 107,65674424042379 Esgut Studio Bandung
-6,951057 107,657524 Jl. Andromeda Tengah II
-6,950901182408997 107,65702567713659 Taman Bima Sakti
-6,947621 107,662547 Jl. Rancabolang
-6,943053955609396 107,66365496968062 Borma Toserba Rancabolang
0 200 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
200 0 140 0 0 0 0 0 0 0 0 0 0 0 0 0 0 62
0 140 0 210 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 210 0 550 0 0 0 400 0 0 0 0 0 0 0 0 0
0 0 0 550 0 390 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 390 0 160 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 160 0 120 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 120 0 550 0 0 0 0 0 0 0 0 0
0 0 0 400 0 0 0 550 0 59 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 59 0 140 0 0 0 0 0 350 0
0 0 0 0 0 0 0 0 0 140 0 140 120 0 350 0 0 0
```

```
0 0 0 0 0 0 0 0 0 0 140 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 120 0 0 120 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 120 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 350 0 0 0 0 56 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 56 0 0 0
0 0 0 0 0 0 0 0 0 350 0 0 0 0 0 0 0 500
0 62 0 0 0 0 0 0 0 0 0 0 0 0 0 0 500 0
```

- Visualisasi Graf



Choose Map: Peta jalan sekitar Buahbatu

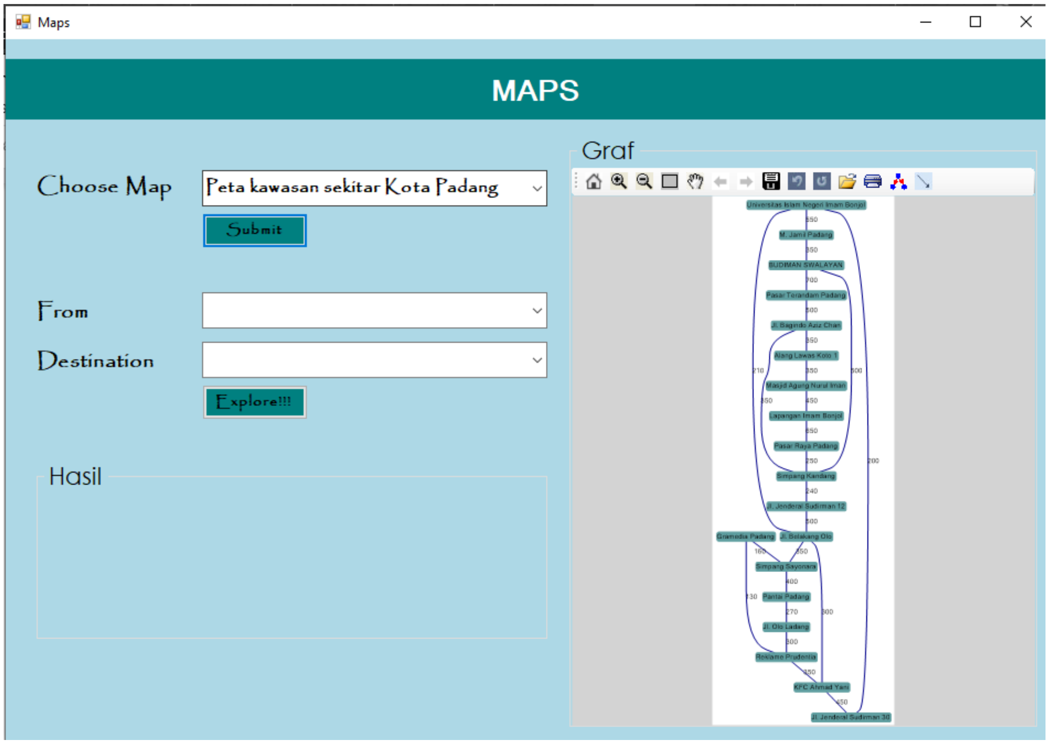


4. Peta kawasan sekitar Kota Padang
    - Input file txt

```
19
-0,942627 100,361987 Jl. Jenderal Sudirman 30
-0,9428719269776579 100,35775407026847 KFC Ahmad Yani
-0,943101 100,354474 Reklame Prudentia
-0,943255 100,351621 Jl. Olo Ladang
-0,9455188136019477 100,3513954403062 Pantai Padang
-0,945735 100,354734 Simpang Sayonara
-0,9441296392028766 100,35442677827504 Gramedia Padang
-0,945740 100,357939 Jl. Belakang Olo
-0,945750 100,362488 Jl. Jenderal Sudirman 12
-0,948143 100,363014 Simpang Kandang
-0,9484322485867474 100,36088956161421 Pasar Raya Padang
-0,9522463448553995 100,36240821756401 Lapangan Imam Bonjol
-0,9544824778208026 100,36237014103814 Masjid Agung Nurul Iman
-0,954440 100,363969 Alang Lawas Koto 1
-0,951339 100,363743 Jl. Bagindo Aziz Chan
-0,9506587199905072 100,36804686746453 Pasar Terandam Padang
-0,9459860471173455 100,36707250860648 BUDIMAN SWALAYAN
-0,943656145453466 100,36665788663193 M. Jamil Padang
-0,9440845360940394 100,36206827670506 Universitas Islam Negeri Imam Bonjol
```

```
0 450 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 200
450 0 350 0 0 0 0 300 0 0 0 0 0 0 0 0 0 0 0
0 350 0 300 0 0 130 0 0 0 0 0 0 0 0 0 0 0 0
0 0 300 0 270 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 270 0 400 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 400 0 160 350 0 0 0 0 0 0 0 0 0 0 0
0 0 130 0 0 160 0 0 0 0 0 0 0 0 0 0 0 0 0
0 300 0 0 0 350 0 0 500 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 500 0 240 0 0 0 0 0 0 0 0 210
0 0 0 0 0 0 0 0 240 0 250 0 0 0 350 0 500 0 0
0 0 0 0 0 0 0 0 0 250 0 650 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 650 0 450 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 450 0 350 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 350 0 350 0 0 0 0
0 0 0 0 0 0 0 0 0 350 0 0 0 350 0 500 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 500 0 700 0 0
0 0 0 0 0 0 0 0 0 500 0 0 0 0 0 700 0 350 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 350 0 550
200 0 0 0 0 0 0 0 210 0 0 0 0 0 0 0 0 0 550 0
```

- Visualisasi Graf

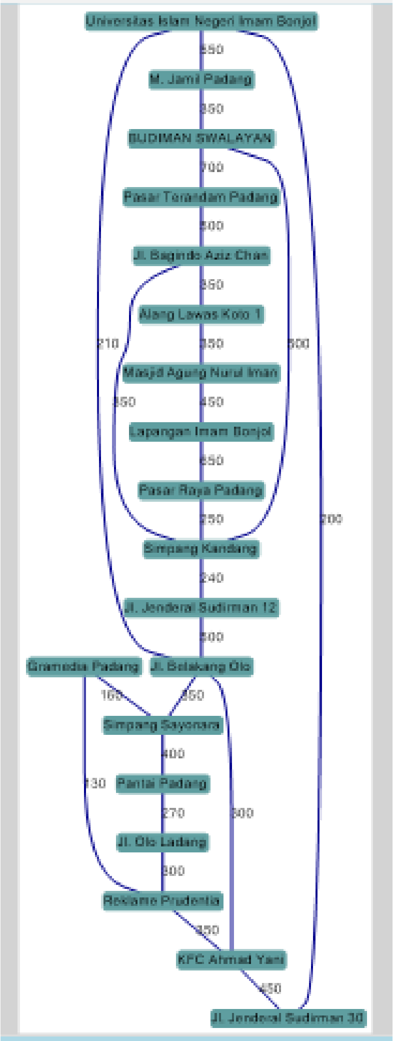


Choose Map: Peta kawasan sekitar Kota Padang

From:

Destination:

Hasil

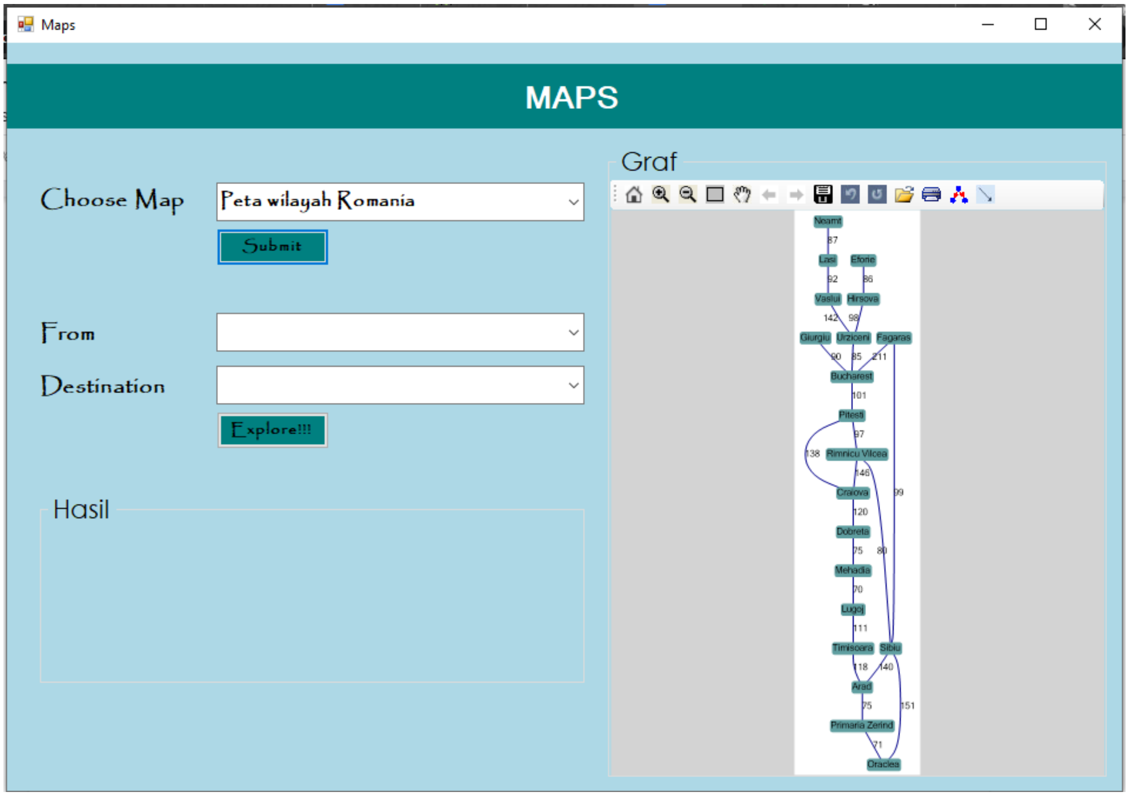


5. Peta wilayah Romania
    - Input file txt

```
20
44,517331381684 26,106035917152663 Oraclea
46,62568544357156 21,51702404001682 Primaria Zerind
46,18635211540256 21,313565330433896 Arad
45,79303068217979 24,12198300567573 Sibiu
45,75565719625707 21,230320297018267 Timișoara
45,69096611262767 21,90304695721961 Lugoj
44,90504713594226 22,367600368550924 Mehadia
44,63691868919327 22,659839495209653 Dobreta
44,33689347987028 23,773360076586346 Craiova
45,09952330258679 24,368276029220638 Ramnicu Valcea
44,85654925603789 24,869307331493864 Pitești
44,46101379069164 26,111383271402115 Bucharest
43,903807280581695 25,969381220096405 Giurgiu
45,841000104771474 24,971912458639125 Fagaras
```

```
44,71802462095263 26,6457071144657 Urziceni
44.68903498229385 27,945485377056507 Hirsova
44,049174806085986 28,652855706687085 Eforie
46,640417812339535 27,72574620249671 Vaslui
47,157907944758534 27,602287646689913 Lasi
46,92789250857661 26,404987648383045 Neamț
0 71 0 151 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
71 0 75 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 75 0 140 118 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
151 0 140 0 0 0 0 0 0 80 0 0 0 99 0 0 0 0 0 0
0 0 118 0 0 111 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 111 0 70 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 70 0 75 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 75 0 120 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 120 0 146 138 0 0 0 0 0 0 0 0 0
0 0 0 80 0 0 0 0 146 0 97 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 138 97 0 101 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 101 0 90 211 85 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 90 0 0 0 0 0 0 0 0
0 0 0 99 0 0 0 0 0 0 0 211 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 85 0 0 0 98 0 142 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 98 0 86 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 86 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 142 0 0 0 92 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 92 0 87
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 87 0
```

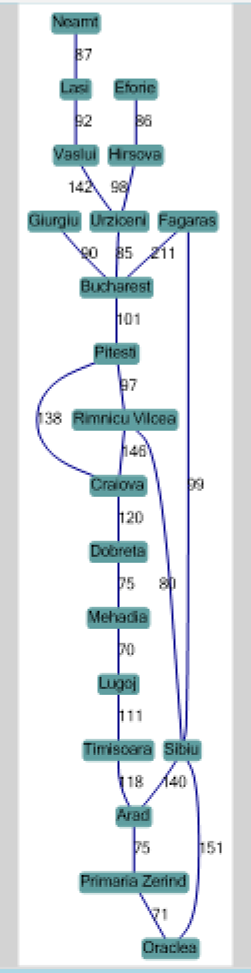
- Visualisasi Graf



Choose Map: Peta wilayah Romania

From:

Destination:

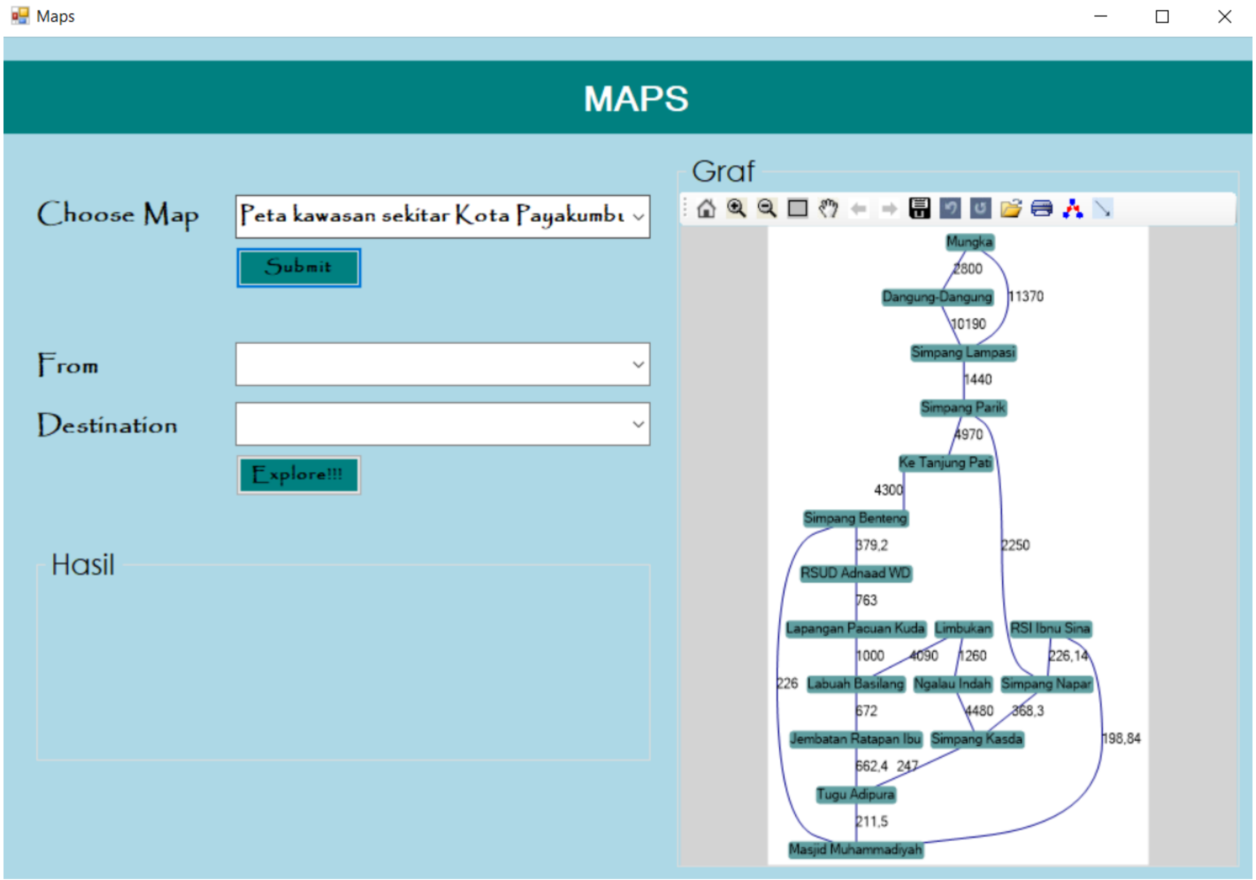


6. Peta kawasan sekitar Kota Payakumbuh
   - Input file txt

```
17
-0,22360052491608298 100,63345821276948 Masjid Muhammadiyah
-0,224544392305494 100,63183127095417 Tugu Adipura
-0,2295493722715176 100,63553846264077 Jembatan Ratapan Ibu
-0,23436806103397195 100,6389786371667 Labuah Basilang
-0,22704367253407415 100,64388828336631 Lapangan Pacuan Kuda
-0,22314951818584902 100,6385546470526 RSUD Adnaad WD
-0,22266114489986222 100,6354066909156 Simpang Benteng
-0,22548316329545343 100,62998829236432 Simpang Kasda
-0,22270878499454003 100,63044940790309 Simpang Napar
-0,22213574166259833 100,63238814041758 RSI Ibnu Sina
-0,19177843810394823 100,65226168464903 Ke Tanjung Pati
-0,20663110169868001 100,61850274587522 Simpang Parik
-0,19480456869931048 100,61381791456736 Simpang Lampasi
-0,13450030056250573 100,54551226185252 Dangung-Dangung
-0,11622500123663988 100,56102444086078 Mungka
-0,25801467388464755 100,6087853250524 Ngalau Indah
```

```
-0,26289376905390915 100,61665387058663 Limbukan
1 )0 211,5 0 0 0 0 226 0 0 198,84 0 0 0 0 0 0 0
2 )211,5 0 662,4 0 0 0 0 247 0 0 0 0 0 0 0 0 0
3 )0 662,4 0 672 0 0 0 0 0 0 0 0 0 0 0 0 0
4 )0 0 672 0 1000 0 0 0 0 0 0 0 0 0 0 0 4090
5 )0 0 0 1000 0 763 0 0 0 0 0 0 0 0 0 0 0
6 )0 0 0 0 763 0 379,2 0 0 0 0 0 0 0 0 0 0
7 )226 0 0 0 0 379,2 0 0 0 0 4300 0 0 0 0 0 0
8 )0 247 0 0 0 0 0 0 368,3 0 0 0 0 0 0 4480 0
9 )0 0 0 0 0 0 0 368,3 0 226,14 0 2250 0 0 0 0 0
10)198,84 0 0 0 0 0 0 0 226,14 0 0 0 0 0 0 0 0
11)0 0 0 0 0 0 4300 0 0 0 0 4970 0 0 0 0 0
12)0 0 0 0 0 0 0 0 2250 0 4970 0 1440 0 0 0 0
13)0 0 0 0 0 0 0 0 0 0 0 1440 0 10190 11370 0
14)0 0 0 0 0 0 0 0 0 0 0 0 10190 0 2800 0 0
15)0 0 0 0 0 0 0 0 0 0 0 0 11370 2800 0 0 0
16)0 0 0 0 0 0 0 4480 0 0 0 0 0 0 0 0 1260
17)0 0 0 4090 0 0 0 0 0 0 0 0 0 0 0 1260 0
```

- Visualisasi Graf



Maps

MAPS

Choose Map: Peta kawasan sekitar Kota Payakumbu

Submit

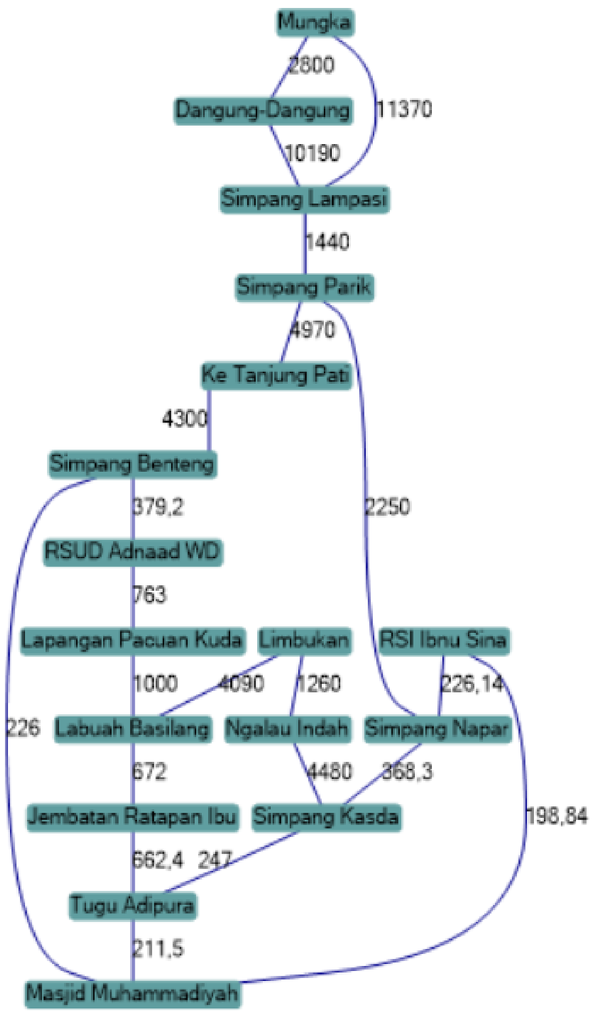
From

Destination

Explore!!!

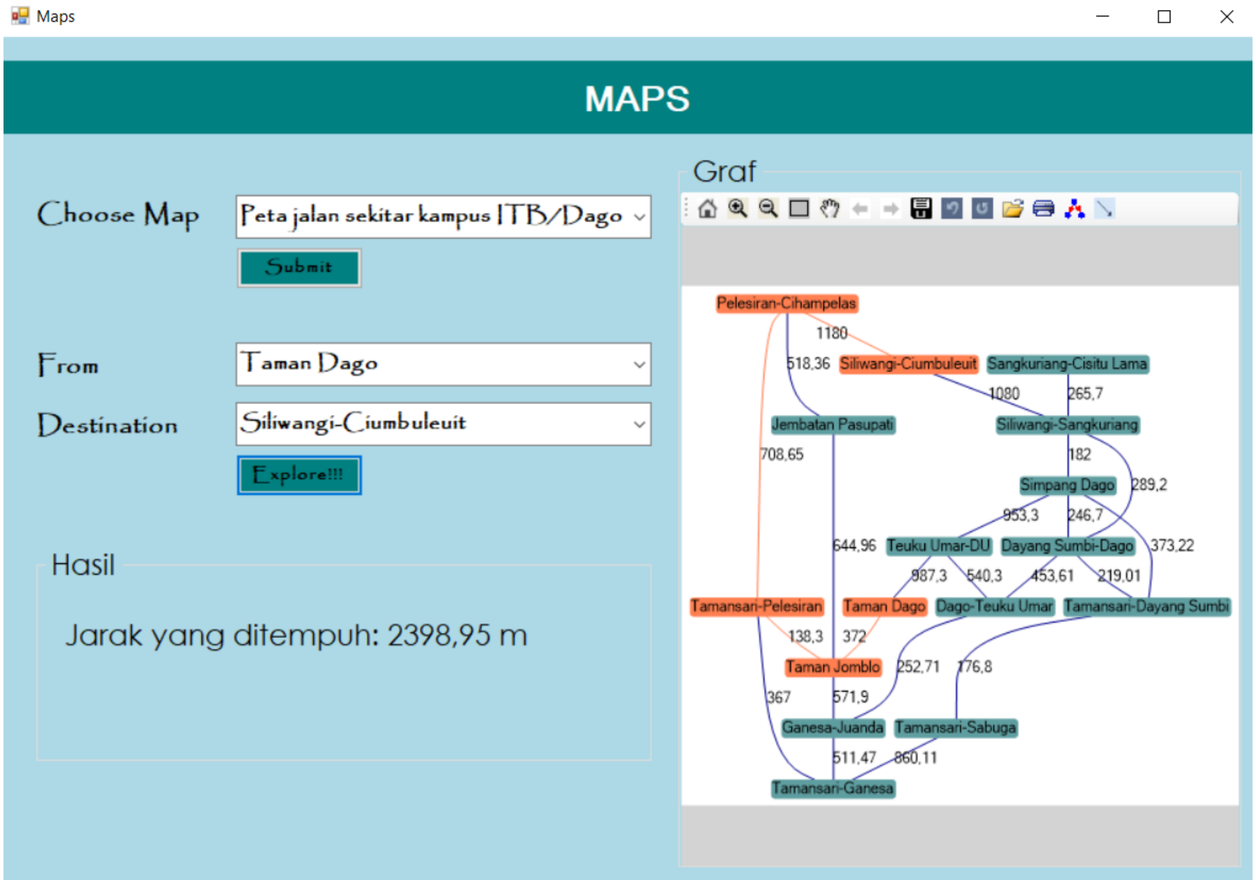
Hasil

Graf



C. Screenshot Hasil Pencarian Rute

1. Peta jalan sekitar kampus ITB/Dago
    - Asal : Taman Dago, Tujuan :Siliwangi-Ciumbuleuit



Choose Map: Peta jalan sekitar kampus ITB/Dago

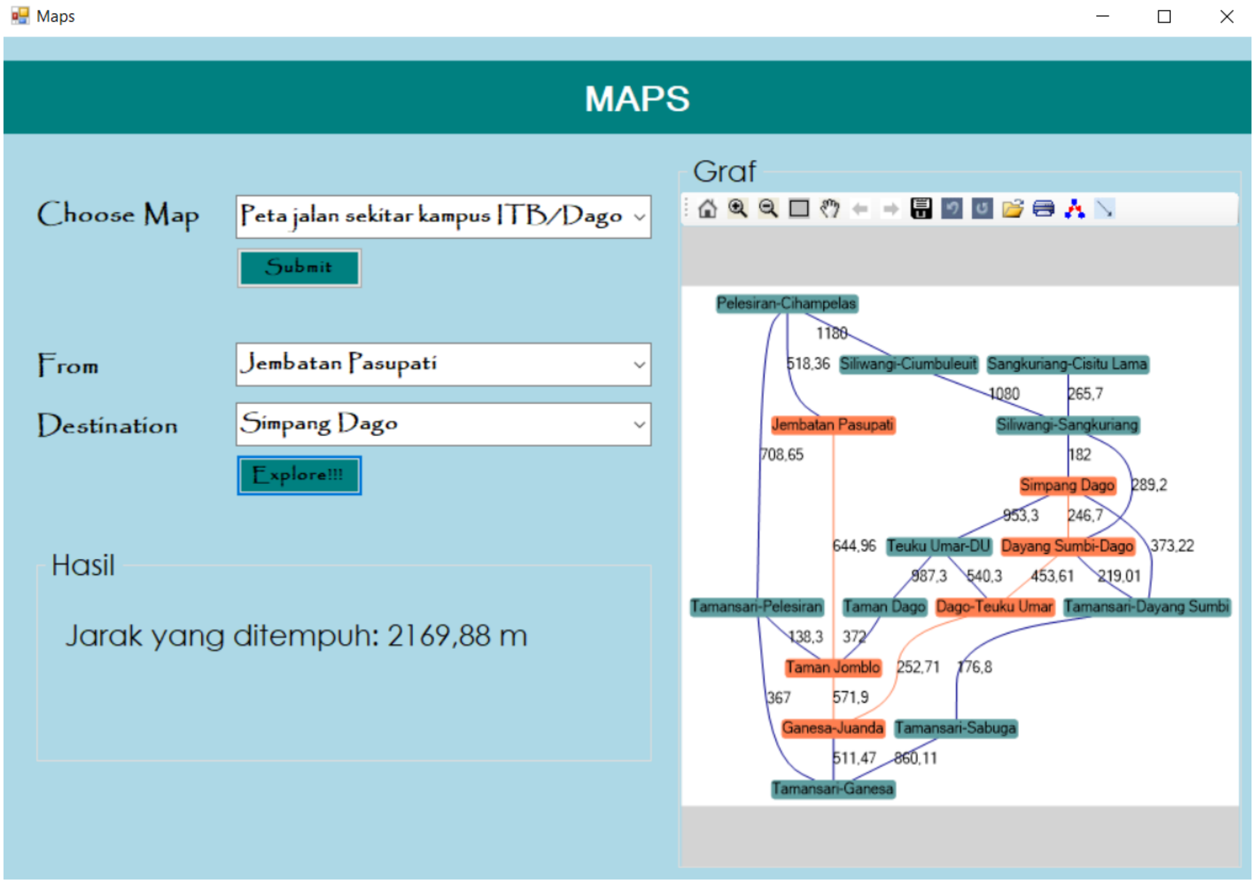
From: Taman Dago

Destination: Siliwangi-Ciumbuleuit

Hasil

Jarak yang ditempuh: 2398,95 m

- Asal : Jembatan Pasupati, Tujuan : Simpang Dago



Choose Map: Peta jalan sekitar kampus ITB/Dago

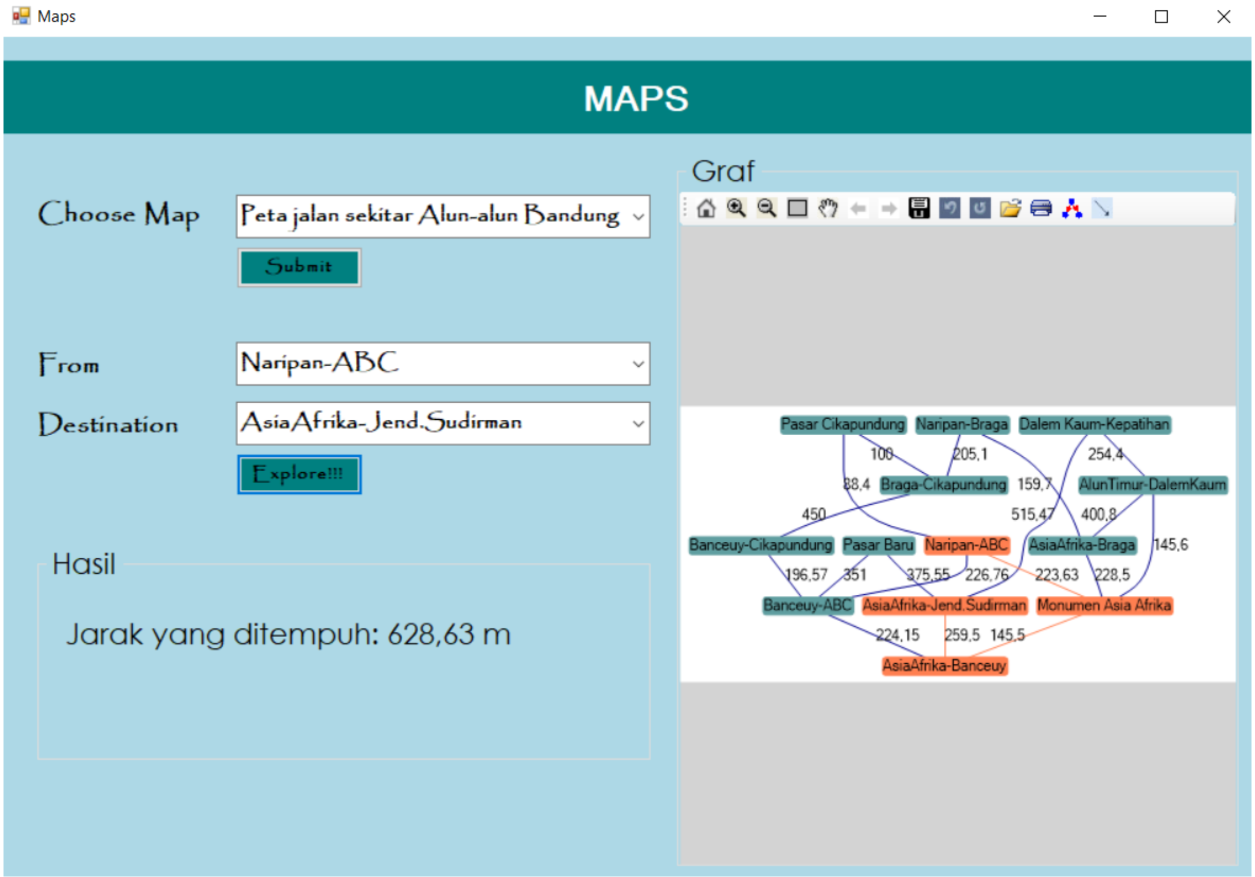
From: Jembatan Pasupati

Destination: Simpang Dago

Hasil

Jarak yang ditempuh: 2169,88 m

2. Peta jalan sekitar Alun-alun Bandung
    - Asal : Naripan-ABC, Tujuan :Asia Afrika-Jend.Sudirman


Maps

MAPS

Choose Map: Peta jalan sekitar Alun-alun Bandung

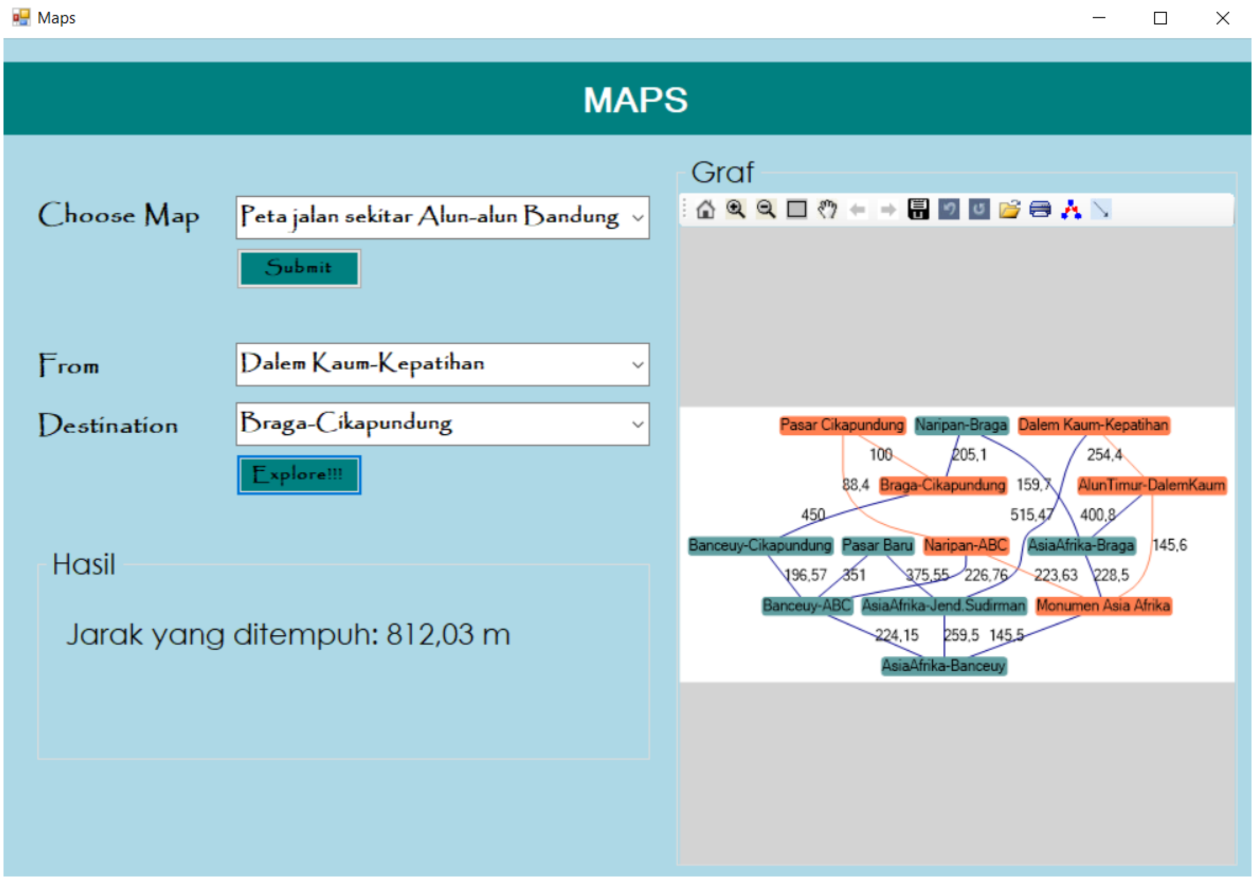
From: Naripan-ABC

Destination: AsiaAfrika-Jend.Sudirman

Hasil

Jarak yang ditempuh: 628,63 m

- Asal : Dalem Kaum-Kepatihan, Tujuan :Braga-Cikapundung



Choose Map: Peta jalan sekitar Alun-alun Bandung

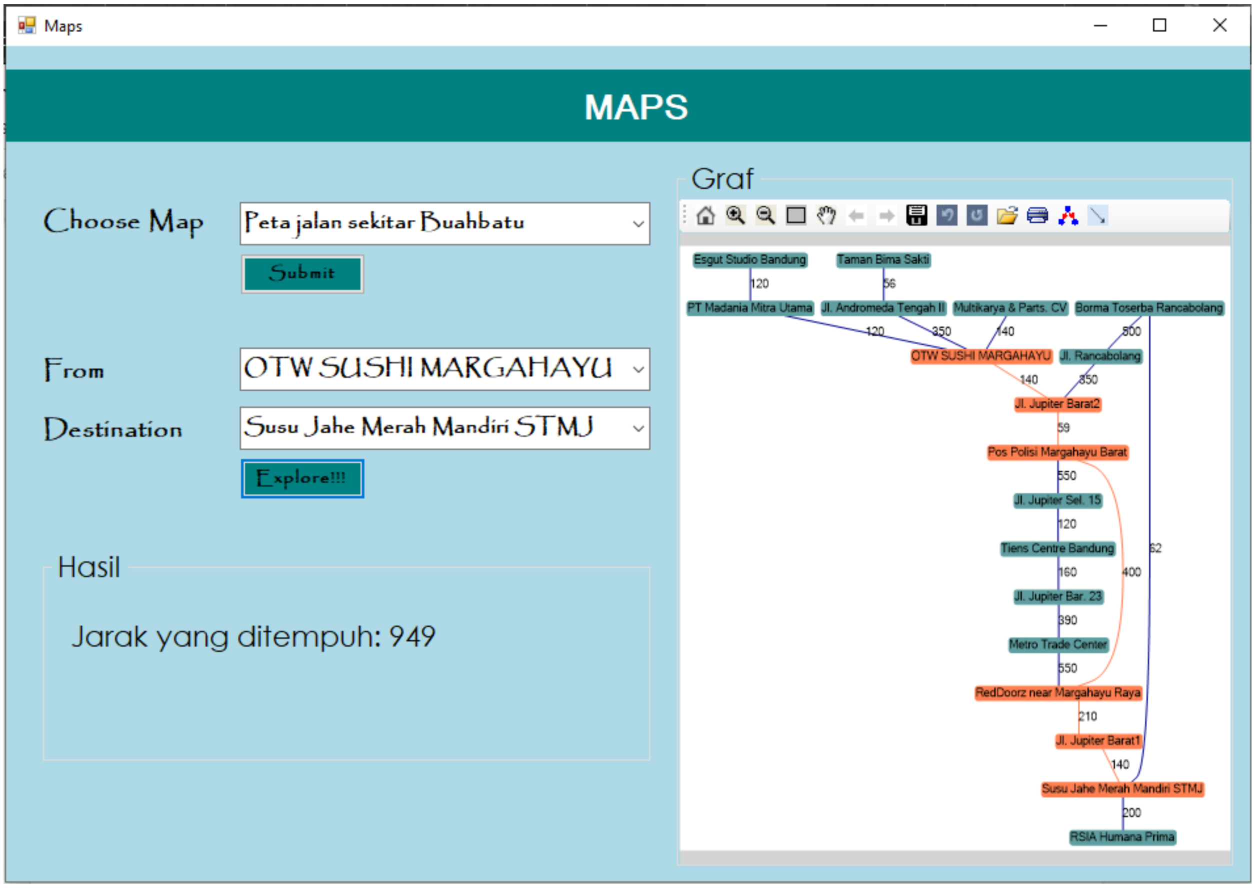
From: Dalem Kaum-Kepatihan

Destination: Braga-Cikapundung

Hasil

Jarak yang ditempuh: 812,03 m

3. Peta jalan sekitar Buahbatu
   - Asal : OTW SUSHI MARGAHAYU, Tujuan : Susu Jahe Merah Mandiri STMJ



Maps

MAPS

Choose Map: Peta jalan sekitar Buahbatu

Submit

From: OTW SUSHI MARGAHAYU

Destination: Susu Jahe Merah Mandiri STMJ

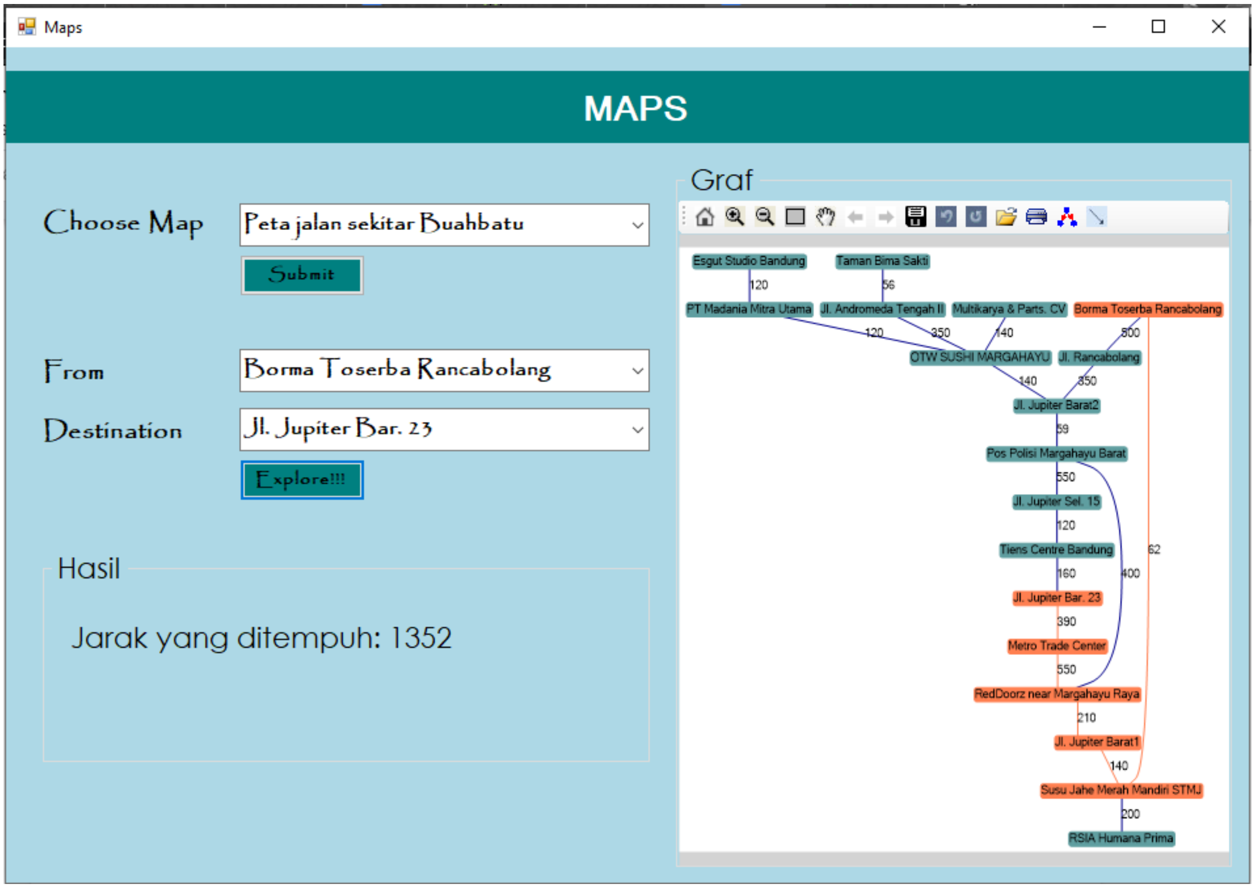
Explore!!!

Hasil

Jarak yang ditempuh: 949

Graf

- Asal : Borma Toserba Rancabolang, Tujuan : Jl. Jupiter Bar. 23



Maps

MAPS

Choose Map: Peta jalan sekitar Buahbatu

Submit

From: Borma Toserba Rancabolang

Destination: Jl. Jupiter Bar. 23

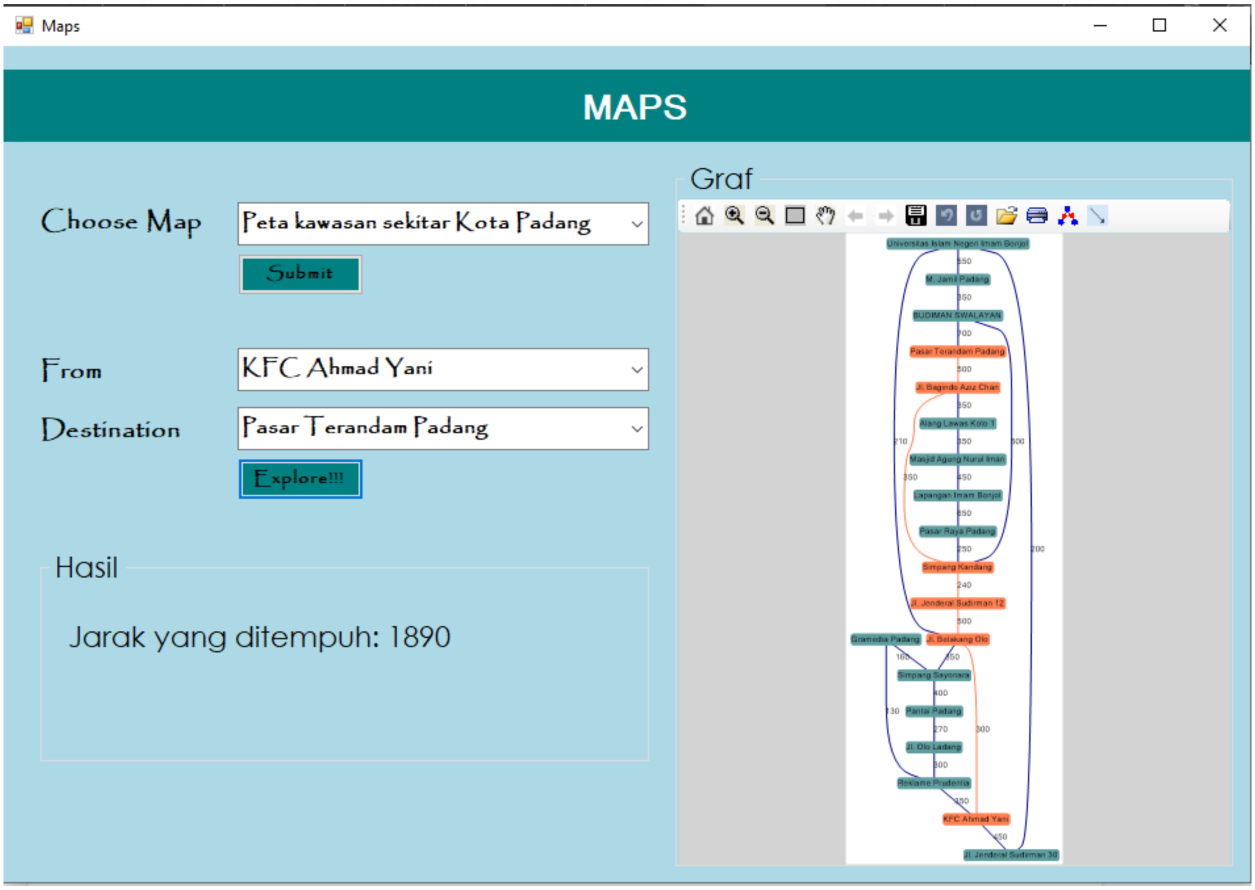
Explore!!!

Hasil

Jarak yang ditempuh: 1352

Graf

4. Peta jalan sekitar kawasan Kota Padang
    - Asal : KFC Ahmad Yani, Tujuan : Pasar Terandam Padang



Choose Map: Peta kawasan sekitar Kota Padang

Submit

From: KFC Ahmad Yani

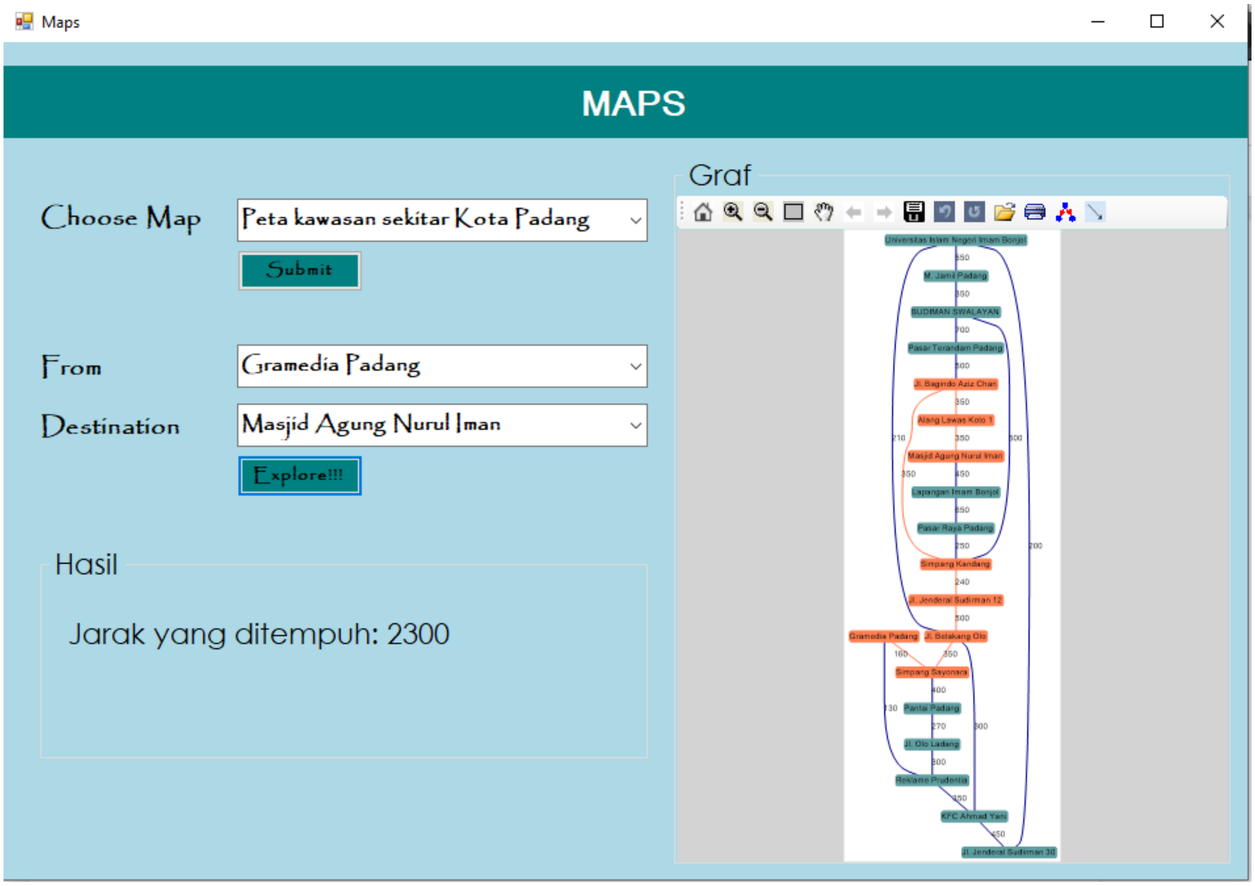
Destination: Pasar Terandam Padang

Explore!!!

Hasil

Jarak yang ditempuh: 1890

- Asal : Gramedia Padang, Tujuan : Masjid Agung Nurul Iman



Choose Map: Peta kawasan sekitar Kota Padang

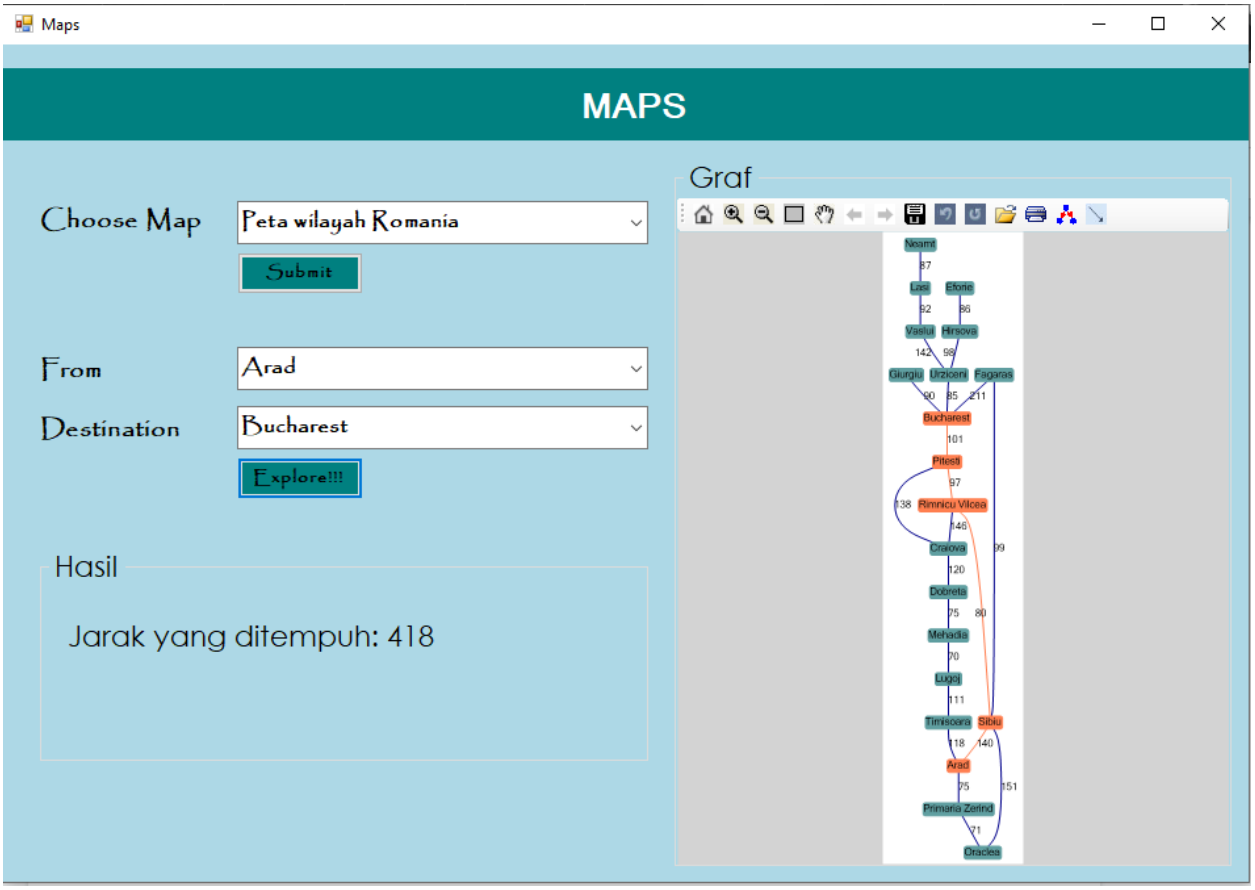
From: Gramedia Padang

Destination: Masjid Agung Nurul Iman

Hasil

Jarak yang ditempuh: 2300

5. Peta wilayah Romania
   - Asal : Arad, Tujuan : Bucharest



Maps

MAPS

Choose Map: Peta wilayah Romania

Submit

From: Arad

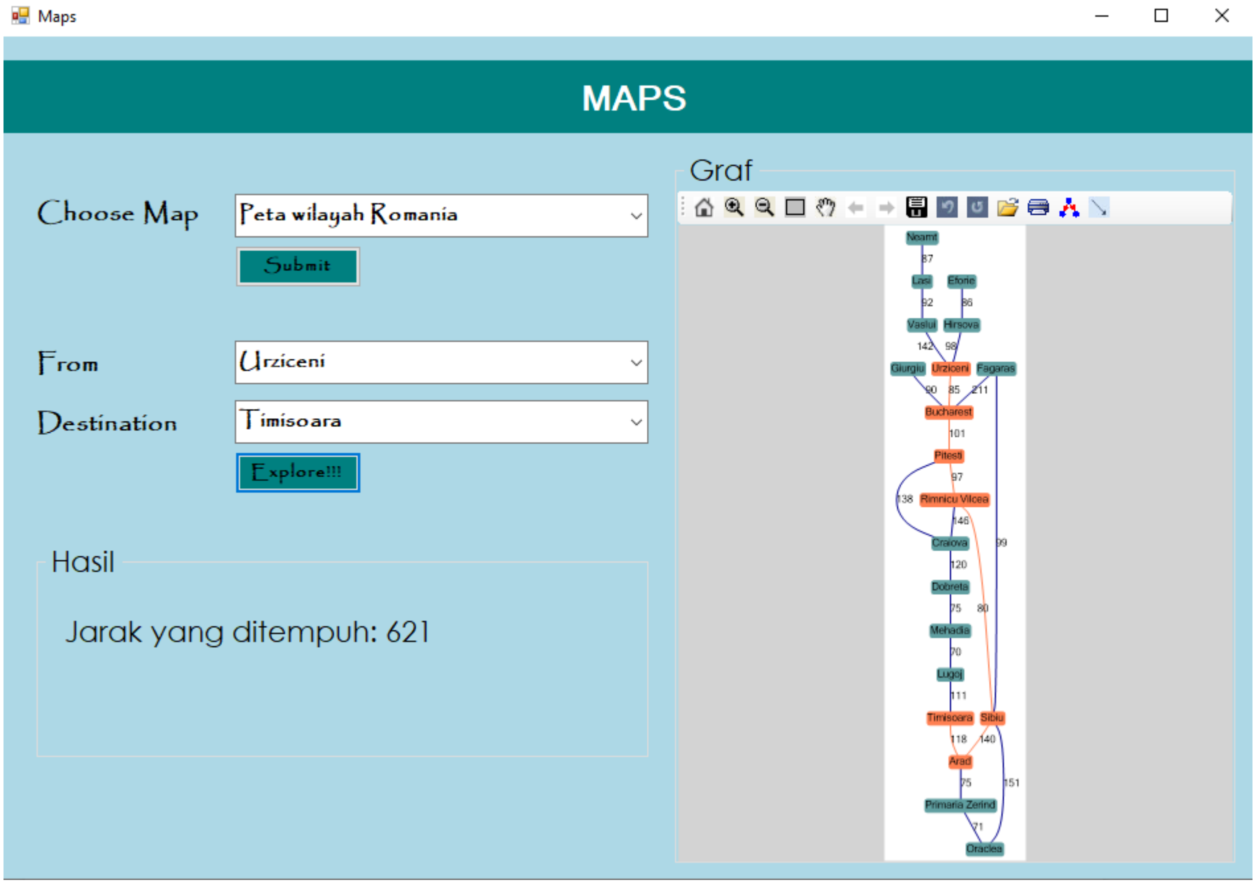
Destination: Bucharest

Explore!!!

Hasil

Jarak yang ditempuh: 418

- Asal : Urziceni, Tujuan : Timisoara



Choose Map: Peta wilayah Romania

Submit

From: Urziceni

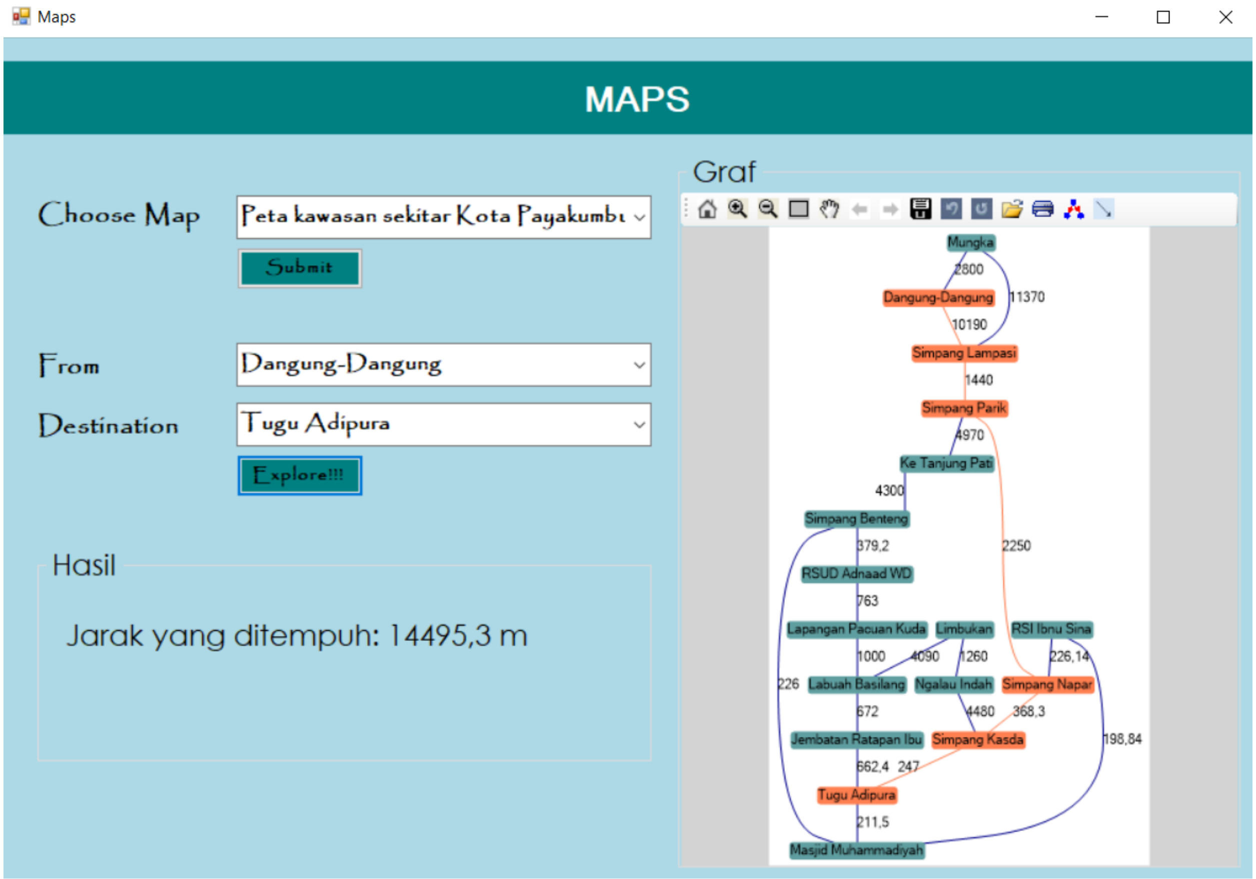
Destination: Timisoara

Explore!!!

Hasil

Jarak yang ditempuh: 621

6. Peta kawasan sekitar Kota Payakumbuh
   - Asal : Dangung-Dangung, Tujuan :Tugu Adipura



Choose Map: Peta kawasan sekitar Kota Payakumbu

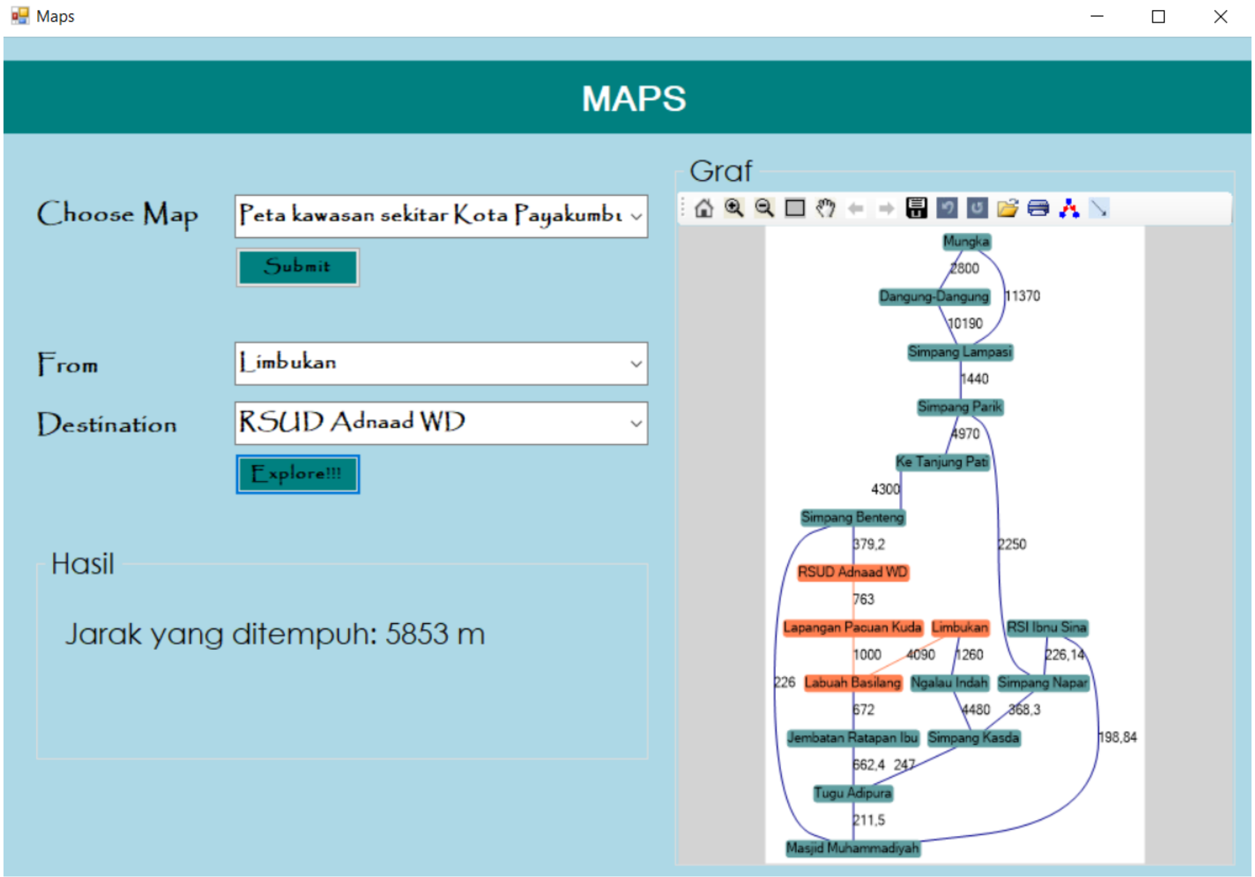
From: Dangung-Dangung

Destination: Tugu Adipura

Hasil

Jarak yang ditempuh: 14495,3 m

- Asal :Limbukan , Tujuan : RSUD Adnaan WD



Maps

MAPS

Choose Map: Peta kawasan sekitar Kota Payakumb

From: Limbukan

Destination: RSUD Adnaad WD

Hasil

Jarak yang ditempuh: 5853 m

Graf

TABEL PENILAIAN

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	