LAPORAN TUGAS KECIL III IF2211 STRATEGI ALGORITMA

Implementasi Algoritma A* untuk Menentukan Lintasan Terpendek

Oleh:

Gregorius Dimas Baskara / 13519190

Tanur Rizaldi Rahardjo / 13519214

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung

2021

1	Program dapat menerima input graf	v
2	Program dapat menghitung lintasan terpendek	v
3	Program dapat menampilkan lintasan terpendek serta jaraknya	v
4	Bonus : Program dapat menerima input peta dengan Google Map API dan menampilkan peta	x

1. Source code

```
// Pathfinding
while (currentLocationName != currentTargetFriend) {

    // Creating branch only if not backtracking
    // Get sorted distance and put to choice stack
    if (!isBacktracking) {
        // List carrying tuple of target location name, distance to, and heuristic value
        List<Tuple<string,float,float>> distanceList = new List<Tuple<string,float,float>>();
        for (int i = 0; i < nameList.Length; i++) {
            if (currentLocationIndex != i && !visited[nameList[i]] && adjacencyMatrix[currentLocationIndex,i] >= 0)
                distanceList.Add(new Tuple<string, float, float> (nameList[i], adjacencyMatrix[currentLocationIndex,i],
                    GetStraightDistance(currentTargetFriend, nameList[i])));
        }
        // Sorting list with Linq, ascending order
        List<Tuple<string,float,float>> sortedDistance = distanceList.OrderBy(obj=>obj.Item3).ToList();

        // Creating available path queue from sorted list
        Queue<Tuple<string,float>> AvailableBranch = new Queue<Tuple<string,float>>();
        foreach (var entry in sortedDistance) {
            AvailableBranch.Enqueue(new Tuple<string,float>(entry.Item1, entry.Item2));
        }

        // Push available path queue to choice stack
        ChoiceStack.Push(AvailableBranch);
    }
```

```
    // Move taking
    if (ChoiceStack.Count != 0) {
        // If choice stack is not exhausted
        Queue<Tuple<string,float>> TopMostBranch = ChoiceStack.Peek();
        if (TopMostBranch.Count != 0) {
            // If choice queue in choice stack is not empty,
            // Move to that location
            currentLocationName = TopMostBranch.Peek().Item1;
            currentLocationIndex = GetIndexFromNameList(currentLocationName);
            float currentPathWeight = TopMostBranch.Peek().Item2;
            TotalPathWeight += currentPathWeight;
            TopMostBranch.Dequeue();

            isBacktracking = false;
            // | Trying new path, so algorithm is stopped backtracking
            CurrentTraversedRoute.Push(new Tuple<string,float>(currentLocationName, currentPathWeight));
            // | Put selected path to route stack
            visited[currentLocationName] = true;
            // | Flagging location as visited
        }
        else {
            // If choice queue is empty, pop choice stack and backtrack
            ChoiceStack.Pop();
            isBacktracking = true;
            // | Set mode to backtracking
            visited[currentLocationName] = false;
```

```
            }
            else {
                // If choice queue is empty, pop choice stack and backtrack
                ChoiceStack.Pop();
                isBacktracking = true;
                // | Set mode to backtracking
                visited[currentLocationName] = false;
                // | Backtracking, removing old path visited flags
                // string LastLocation = CurrentTraversedRoute.Peek(); // DEBUG
                TotalPathWeight -= CurrentTraversedRoute.Peek().Item2;
                CurrentTraversedRoute.Pop();
                // | Remove last path from route stack
            }
        }
        else {
            // If choice stack is exhausted, then no path found
            System.Windows.Forms.MessageBox.Show("Path not found", "Alert");
            isSolutionFound = false;
            break;
        }

    }
```

2. **Input file**

```
Monkasel -1 830.12 266.68 -1 -1 -1 -1 -1
Grand_city 830.12 -1 492.83 -1 -1 -1 594.66 -1
Stasiun_Gubeng 266.68 492.83 -1 -1 979.18 -1 -1 -1
Masjid_Cheng_Hoo -1 -1 -1 -1 -1 1860 1210 -1
RS_Dr_Sutomo -1 -1 979.18 -1 -1 876.93 -1 -1
Pasar_Pacarkeling -1 -1 -1 1860 876.93 -1 -1 -1
Sate_Kelopo_Ondomohen -1 594.66 -1 1210 -1 -1 -1 1150
Tunjungan -1 -1 -1 -1 -1 -1 -1 1150
Monkasel -7.26544 112.75028
Grand_city -7.26193 112.74997
Stasiun_Gubeng -7.26546 112.75200
Masjid_Cheng_Hoo -7.25199 112.74717
RS_Dr_Sutomo -7.26793 112.75676
Pasar_Pacarkeling -7.25995 112.75909
Sate_Kelopo_Ondomohen -7.26011 112.74419
Tunjungan -7.25612 112.73778
```

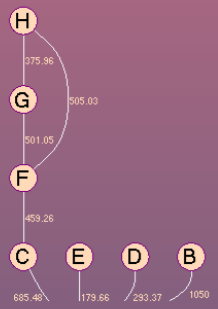
3. **Lintasan terpendek**

MANGGA

Track Where Ever You Go!

TRACK GRAPH

Browse



Legend

A : Pizza_Hut_Buahbatu
B : RSAI_Buahbatu
C : Mandiri_University
D : Metro_Indah_Mall

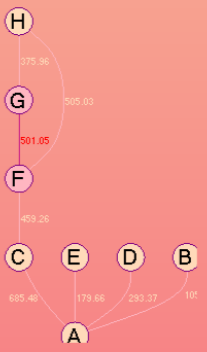
Where Are You?

Edelweiss Hospit

Where To Go?

Alfamart

Search



Result

Edelweiss_Hospital - Alfamart

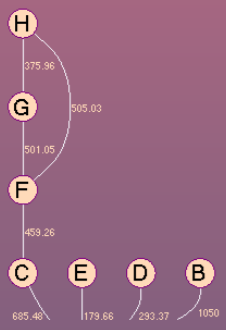
Total distance : 501.05

MANGGA

Track Where Ever You Go!

TRACK GRAPH

Browse



Legend

A : Pizza_Hut_Buahbatu
B : RSAI_Buahbatu
C : Mandiri_University

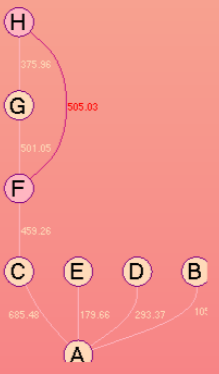
Where Are You?

Edelweiss Hospit

Where To Go?

KPU Bandung

Search



Result

Edelweiss_Hospital - KPU_Bandung

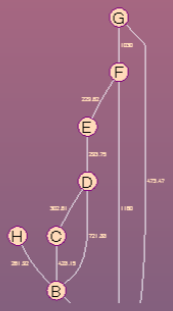
Total distance : 505.03

MANGGA

Track Where Ever You Go!

TRACK GRAPH

Browse



Legend

A : ITB_Belakang
B : Jl_Ir_H_Juanda-Dago
C : Kantor_Cabang_BCA
D : RS_Borromeus

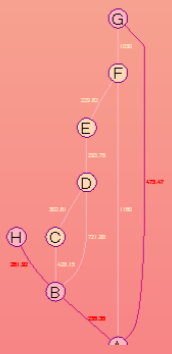
Where Are You?

Sabuga

Where To Go?

Mc Donald Dago

Search



Result

Sabuga - ITB_Belakang - Jl_Ir_H_Juanda-Dago - Mc_Donald_Dago

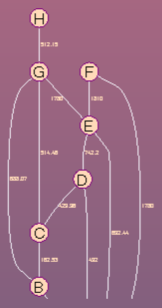
Total distance : 970.73

MANGGA

Track Where Ever You Go!

TRACK GRAPH

Browse



Legend

A : Alun_Alun
B : Bank_Mandiri_Asia_Afrika
C : Wings_o_wings
D : Taman_Braga
E : Hotel_Pasar_Baru

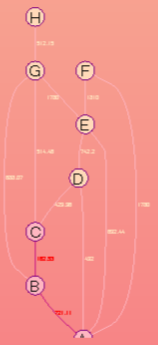
Where Are You?

Wings o wings

Where To Go?

Alun Alun

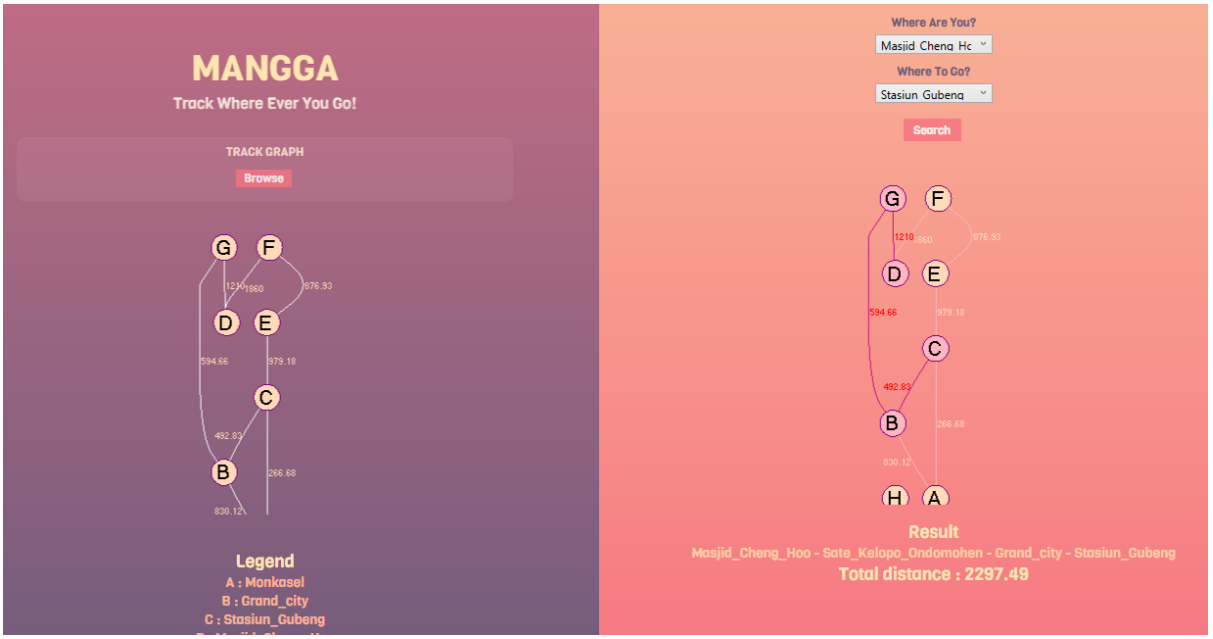
Search



Result

Wings_o_wings - Bank_Mandiri_Asia_Afrika - Alun_Alun

Total distance : 883.64



4. Link Repository

[Repository Tugas Kecil 3]