

LAPORAN
TUGAS BESAR 1 STRATEGI ALGORITMA
MINIMUM SPANNING TREE



Disusun Oleh :

- | | |
|-----------------------------------|-------------------|
| 1. Dzaky Dhimast Rusdianto | (14116041) |
| 2. Bahrul Faizi | (14116037) |
| 3. Meiji Suryadi | (14116171) |
| 4. Muhammad Julian | (14116165) |

PROGRAM STUDI TEKNIK INFORMATIKA
INSTITUT TEKNOLOGI SUMATERA

LAPORAN TUGAS BESAR

Disusun untuk memenuhi persyaratan pengumpulan tugas besar mata kuliah Strategi
Algoritma program studi Teknik Informatika - Institut Teknologi Sumatera
tahun ajaran 2017/2018

Disusun Oleh :

- | | |
|-----------------------------------|-------------------|
| 1. Dzaky Dhimast Rusdianto | (14116041) |
| 2. Bahrul Faizi | (14116037) |
| 3. Meiji Suryadi | (14116171) |
| 4. Muhammad Julian | (14116165) |

**PROGRAM STUDI TEKNIK INFORMATIKA
INSTITUT TEKNOLOGI SUMATERA
2018**

KATA PENGANTAR

Assalamu'alaikumWr. Wb.

Puji dan syukur penulis panjatkan kehadirat Allah SWT karena telah memberikan rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan laporan tugas besar mata kuliah Strategi Algoritma tahun ajaran semester genap Institut Teknologi Sumatera.

Dalam kesempatan ini penulis ingin mengucapkan banyak terima kasih kepada yang terhormat Bapak Imam Eko Wicaksono selaku dosen pengampu mata kuliahStrategi Algoritma, kepada semua rekan yang telah membantu kami dalam pengerjaan laporan ini, dan tak lupa kepada kedua orang tua dan seluruh keluarga tercinta kami bertiga yang tentunya telah memberikan do'a dan dorongan baik secara moril maupun secara materiil.

Penulis menyadari bahwa dalam penyusunan laporan ini masih banyak kekurangan, mengingat adanya kemampuan dan pengetahuan yang kami miliki masih sangat terbatas, maka dari itu penulis mengharapkan saran dan kritik demi menyempurnakan laporan tugas besar. Semoga laporan tugas besar akhir semester ini dapat memberikan banyak manfaat bagi banyak orang.

Wassalamu'alaikumWr. Wb.

Lampung Selatan, Mei 2018

Penulis

BAB I

PENDAHULUAN

1.1 Latar Belakang

Algoritma Prim adalah sebuah algoritma dalam teori graf untuk mencari pohon rentang minimum untuk sebuah graf berbobot yang saling terhubung. Ini berarti bahwa sebuah himpunan bagian dari *edge* yang membentuk suatu pohon yang mengandung *node*, di mana bobot keseluruhan dari semua *edge* dalam pohon diminimalisasikan. Bila graf tersebut tidak terhubung, maka graf itu hanya memiliki satu pohon rentang minimum untuk satu dari komponen yang terhubung.

Pada program yang kami buat, Algoritma Prim digunakan untuk mencari rute terpendek dengan biaya minimum untuk mencapai suatu tempat dari tempat yang lain.

1.2 Batasan Masalah

Agar tidak menyimpang dari topik yang akan di bahas maka penulis menekankan batasan masalah yang akan di bahas, hanya terbatas pada :

1. Menggunakan bahasa pemrograman yang dikuasai.
2. Membuat program yang dapat menentukan rute terpendek dengan biaya perjalanan minimum.
3. Membuat program yang dapat menentukan rute terpendek dengan biaya perjalanan minimum dari suatu tempat ke tempat yang lain.
4. Area dibatasi hanya di kecamatan Way Halim, node – node yang digunakan sebagai titik – titik lokasi adalah setiap persimpangan jalan yang ada.
5. Jalan – jalan yang digunakan dibatasi hanya jalan yang memenuhi syarat, yaitu yang sesuai ukuran dapat dilalui 2 buah bus (searah maupun berlawanan arah).

1.3 Tujuan Penelitian

1. Untuk merancang program pohon rentang minimum menggunakan algoritma yang telah dipelajari dengan bahasa pemrograman yang telah dikuasai.
2. Memenuhi tugas besar matakuliah Strategi Algoritma.
3. Penerapan materi dari algoritma Prim.

1.4 Manfaat Program

Setelah semua tujuan telah diuraikan di atas maka manfaat yang bisa diperoleh adalah:

1. Dapat menentukan rute terpendek suatu tempat ke tempat yang lain.
2. Dapat menentukan biaya perjalanan minimum yang dibutuhkan dari satu tempat ke tempat yang lain.

BAB II

METODOLOGI

2.1 Analisis Algoritma

2.1.1 Analisis Program

Program ini dibuat berdasarkan teori – teori yang sudah dipelajari pada matakuliah Strategi Algoritma, khususnya algoritma Prim yang sudah dijelaskan. Pada dasarnya harus memiliki kriteria sebagai berikut :

- Membuat program yang dapat menentukan rute terpendek dengan biaya perjalanan minimum dari suatu tempat ke tempat yang lain.
- Area dibatasi hanya di kecamatan Way Halim, node – node yang digunakan sebagai titik – titik lokasi adalah setiap persimpangan jalan yang ada.
- Jalan – jalan yang digunakan dibatasi hanya jalan yang memenuhi syarat, yaitu yang sesuai ukuran dapat dilalui 2 buah bus (searah maupun berlawanan arah).

2.1.2 Graf

Ilustrasi peta:



14 Node

1 - 4	: 4000	: 2,6 km
1 - 2	: 4000	: 1,1 km
2 - 3	: 4000	: 1,1 km
3 - 4	: 4000	: 400 m
3 - 7	: 4000	: 1 km
7 - 14	: 4000	: 1,6 km
7 - 10	: 4000	: 1,4 km
10 - 13	: 4000	: 300 m
13 - 14	: 4000	: 800 m
13 - 12	: 4000	: 1,6 km
12 - 11	: 4000	: 800 m
11 - 10	: 4000	: 1 km
11 - 9	: 4000	: 600 m
9 - 8	: 4000	: 600 m
8 - 5	: 4000	: 1,4 km
5 - 6	: 4000	: 500 m
6 - 2	: 4000	: 700 m
5 - 7	: 4000	: 1,8 km

Graf yang sudah dijadikan matriks:

graf - Notepad													
File	Edit	Format	View	Help									
14	-1	1100	-1	2600	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	1100	-1	1100	-1	700	-1	-1	-1	-1	-1	-1	-1	-1
-1	1100	-1	400	-1	-1	1000	-1	-1	-1	-1	-1	-1	-1
2600	-1	400	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	500	1800	1400	-1	-1	-1	-1	-1	-1
-1	700	-1	-1	500	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	1000	-1	1800	-1	-1	-1	-1	1400	-1	-1	-1	1600
-1	-1	-1	-1	1400	-1	-1	-1	600	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	600	-1	-1	600	-1	-1	-1
-1	-1	-1	-1	-1	-1	1400	-1	-1	-1	1000	-1	900	-1
-1	-1	-1	-1	-1	-1	-1	-1	600	1000	-1	800	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	800	-1	1600	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	900	-1	1600	-1	800
-1	-1	-1	-1	-1	-1	1600	-1	-1	-1	-1	-1	800	-1

2.1.3 Source Code

```
#include <iostream>
#include <fstream>
#define inf 99999
using namespace std;

int main() {
    int ukuran;
    int x,y,u,v;
    int min;
    int lewati[100]={0};
    int bobot_minimum=0;
    int sisi=1;
    int i=1;
    int j=1;

    ifstream file;

    file.open("graf.txt");

    file>>ukuran;
    int matriks[ukuran][ukuran];

    while (!file.eof())
    {
        for (i=1;i<=ukuran;i++)
        {
            for (j=1;j<=ukuran;j++)
```



```

        {
            file>>matriks[i][j];

            if (matriks[i][j]==-1)
            {
                matriks[i][j]=inf;
            }
        }
    }

}

file.close();

lewati[1]=1;

while(sisi < ukuran)
{

    for(i=1,min=inf;i<=ukuran;i++)

        for(j=1;j<=ukuran;j++)

            if(matriks[i][j]< min)

                if(lewati[i]!=0)

                    {

                        min=matriks[i][j];

```

```

        x=u=i;

        y=v=j;

    }

    if(lewati[u]==0 || lewati[v]==0)

    {
        sisi++;
        cout<<"("<<x<<","<<y<<") = "<<min<<endl;

        bobot_minimum+=min;
        lewati[y]=1;

    }

    matriks[x][y]=matriks[y][x]=inf;

}

cout<<"\nTotal bobot minimum spanning tree = "<<bobot_minimum;

return 0;

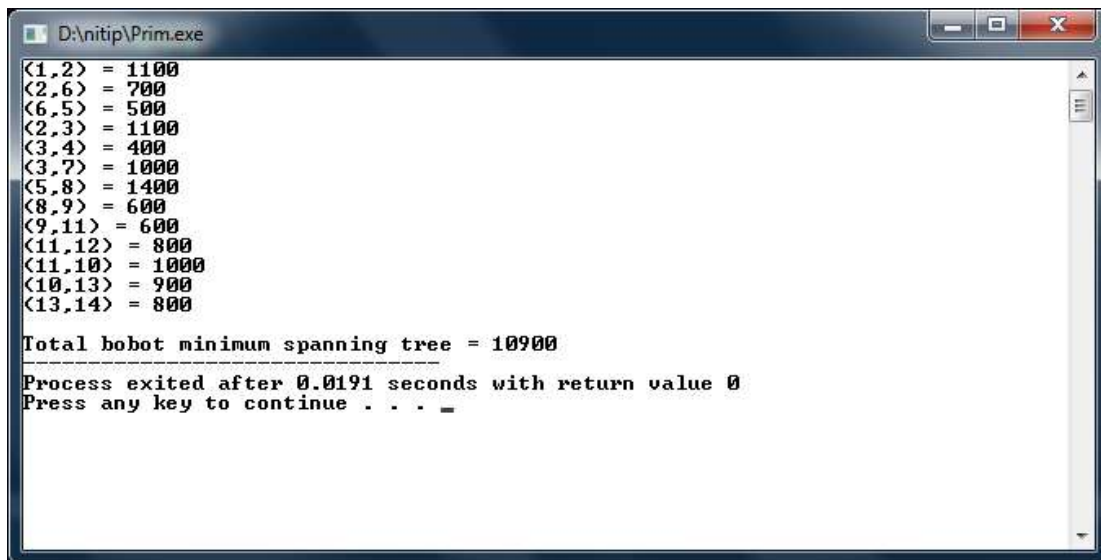
}

```

BAB III HASIL DAN PEMBAHASAN

3.1 Implementasi Program

Berikut merupakan implementasi program :



```
D:\nitip\Prim.exe
<1,2> = 1100
<2,6> = 700
<6,5> = 500
<2,3> = 1100
<3,4> = 400
<3,7> = 1000
<5,8> = 1400
<8,9> = 600
<9,11> = 600
<11,12> = 800
<11,10> = 1000
<10,13> = 900
<13,14> = 800

Total bobot minimum spanning tree = 10900
-----
Process exited after 0.0191 seconds with return value 0
Press any key to continue . . . _
```

Gambar 1. Tampilan Rute dan Biaya Perjalanan Minimum

Pada state ini program akan menampilkan tiap simpul dan berapa biaya perjalanan dari satu simpul ke simpul lainnya. Saat program dijalankan akan menampilkan output seperti pada gambar di atas.

3.2 Graf

Berikut adalah graf yang didapat dari mencari simpul – simpul yang ada pada peta Wayhalim. Simpangan pada jalan dimisalkan sebagai simpul pada graf.



Gambar 2. Tampilan Peta Wayhalim dengan Menggunakan Graf

BAB IV

KESIMPULAN DAN SARAN

4.1 Kesimpulan

Setelah menyelesaikan penyusunan laporan tugas akhir semester ini, penulis dapat menyimpulkan bahwa membuat program ini tidaklah mudah. Tetapi dengan terselesainya program ini kita dapat menggunakannya untuk menghitung. Dari hasil pembuatan program dapat ditarik kesimpulan sebagai berikut :

- Program ini dapat digunakan untuk mencari rute terpendek dari satu tempat ke tempat yang lain.
- Setelah program ini diterapkan pengguna dapat dengan mudah menjalankan tugasnya.
- Dapat meingimplementasikan algoritma Prim ke dalam program sederhana.

4.2 Saran

Berdasarkan hasil kesimpulan, maka dapat diberikan saran-saran yang berguna bagi pengembangan selanjutnya, agar pengembang atau peneliti selanjutnya dapat membuat program ini menjadi lebih baik dan mudah dipahami.

DAFTAR PUSTAKA

Munir, Rinaldi. 2003. *Diktat Kuliah IF2153. Matematika Diskrit Edisi Keempat*. Bandung: Penerbit ITB. <http://www.informatika.org/~rinaldi>.

Kalpanadevi, D. 2013. Effective Searching Shortest Path In Graph Using Prim's Algorithm. *International Journal of Computer & Organization Trends – Volume 3 Issue8 – Sep 2013*