

**PENERAPAN ALGORITMA *FLOYD WARSHALL* DALAM  
PENENTUAN LINTASAN TERPENDEK (*SHORTEST  
PATH*) TERHADAP PENCARIAN DESTINASI  
WISATA DI KOTA PALOPO**

**ALDI WAHYUDI  
1503407017**



**FAKULTAS SAINS  
UNIVERSITAS COKROAMINOTO PALOPO  
2020**

**PENERAPAN ALGORITMA *FLOYD WARSHALL* DALAM  
PENENTUAN LINTASAN TERPENDEK (*SHORTEST  
PATH*) TERHADAP PENCARIAN  
DESTINASI WISATA DI  
KOTA PALOPO**

**SKRIPSI**

Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Sains pada  
Program Studi Matematika Fakultas Sains  
Universitas Cokroaminoto Palopo

**ALDI WAHYUDI  
1503407017**

**PROGRAM STUDI MATEMATIKA  
FAKULTAS SAINS  
UNIVERSITAS COKROAMINOTO PALOPO  
2020**

## PENGESAHAN SKRIPSI

Judul : Penerapan Algoritma *Floyd Warshall* dalam Penentuan Lintasan Terpendek (*Shortest Path*) Terhadap Pencarian Destinasi Wisata Di Kota Palopo  
Nama : Aldi Wahyudi  
NIM : 1503407017  
Program Studi : Matematika  
Tanggal Ujian : 8 Februari 2020

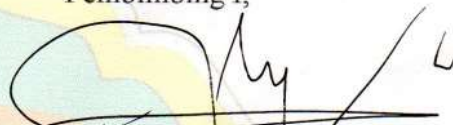
Menyetujui,

Pembimbing II,



Yuliani, S.Si., M.Si.


Pembimbing I,



Drs. Muhammad Ilyas, M.Pd.

Mengesahkan,

Ketua Program Studi Matematika,

  
Marwan Sam, S.Si., M.Si.  
Tanggal: 06/11/2020

Dekan Fakultas Sains,

  
Pauline D. Kasi, S.Si., M.Sc.  
Tanggal: 06/11/2020

## ABSTRAK

**Aldi Wahyudi.** 2020. Penerapan Algoritma *Floyd Warshall* dalam Menentukan Lintasan Terpendek (*Shortest Path*) Terhadap Pencarian Destinasi Wisata di Kota Palopo (dibimbing oleh Muhammad Ilyas dan Yuliani)

Penelitian ini bertujuan untuk menentukan lintasan terpendek (*shortest path*) terhadap pencarian destinasi wisata di Kota Palopo menggunakan Algoritma *Floyd Warshall*. Data yang digunakan dalam penelitian ini adalah data sekunder berupa data jarak antar destinasi wisata di Kota Palopo yang diperoleh dari *Google Maps*. Data yang diperoleh direpresentasikan dalam bentuk graf yang selanjutnya diproses dengan perhitungan manual sesuai langkah-langkah algoritma *Floyd Warshall*. Hasil penelitian menunjukkan jarak terpendek dari semua titik ke semua titik pada destinasi wisata di Kota Palopo. Sebagai contoh, jarak terpendek titik  $v_5$  ke  $v_7$  adalah 14,3 km dengan lintasan  $v_5 \rightarrow v_6 \rightarrow v_4 \rightarrow v_7$ .

Kata kunci: Graf, *Shortest Path* Algoritma *Floyd Warshall*, Destinasi Wisata

## KATA PENGANTAR

*Assalamu'alaikum Wr. Wb*

Puji dan syukur kita panjatkan ke hadirat Tuhan Yang Maha Esa, yang telah melimpahkan rahmat serta hidayah-Nya dan Nabi Muhammad SAW beserta keluarga-Nya yang selama ini telah memberikan penulis kekuatan dan semangat sehingga penyusunan skripsi ini dapat diselesaikan dengan baik dan tepat waktu. Penulisan skripsi ini merupakan salah satu syarat untuk mendapatkan gelar sarjana pada Program S1 Matematika Fakultas Sains Universitas Cokroaminoto Palopo.

Penulis dengan penuh kerendahan hati mengucapkan terimakasih kepada Bapak, Ibu, Orang tua serta saudara-saudariku yang telah memberikan bantuan materi, ridho dan doa yang tiada hentinya untuk penulis dari awal kuliah sampai selesainya penyusunan skripsi ini, kepada adik beserta teman-teman yang telah memberikan semangat dan motivasi kepada penulis, penulis ucapkan terimakasih.

Selama dalam penyusunan skripsi ini penulis telah banyak memperoleh bantuan dan bimbingan, untuk itu penulis ingin mengucapkan terimakasih sebesar-besarnya kepada:

1. Bapak Prof. Hanafie Mahtika, M.S., selaku Rektor Universitas Cokroaminoto Palopo.
2. Ibu Pauline Destinugrainy Kasi, S.Si., M.Sc., selaku Dekan Fakultas Sains Universitas Cokroaminoto Palopo.
3. Bapak Marwan Sam, S.Si., M.Si, selaku Ketua Program Studi Matematika Fakultas Sains yang telah memberikan kelancaran urusan akademik.
4. Bapak Dr. Muhammad Ilyas, M.Pd, selaku Pembimbing I.
5. Ibu Yuliani, S.Si., M.Si., selaku Pembimbing II yang telah banyak serta meluangkan waktunya kepada penulis sampai skripsi ini tersusun.
6. Bapak dan Ibu Dosen Program Studi Matematika Universitas Cokroaminoto Palopo yang telah banyak memberikan saran atau masukan dalam skripsi ini.
7. Orang tua dan keluarga penulis yang selalu memberikan segala dukungan dan motivasi sehingga penulis dapat menyelesaikan skripsi ini.

8. Teman-teman seperjuangan angkatan 2015 yang selalu memberikan bantuan, motivasi, informasi demi tersusunnya skripsi ini.
9. Keluarga besar HIMATIKA SAINS UNCP yang selalu memberikan bantuan, motivasi, saran atau masukan demi tersusunnya skripsi ini.
10. Sahabat-sahabat terdekat penulis yang selalu mengingatkan perkuliahan sekaligus memberikan bantuan, motivasi, informasi serta kerjasama yang baik demi tersusunnya skripsi ini.
11. Semua pihak yang tidak bisa disebutkan satu persatu yang telah membantu dalam penyusunan skripsi ini.

Karena keterbatasan kemampuan, penulis menyadari bahwa penulisan skripsi ini masih jauh dari kesempurnaan. Untuk itu kritik dan saran yang sangat penulis harapkan demi kesempurnaannya. Akhir kata, semoga skripsi ini dapat bermanfaat bagi perkembangan ilmu terkhusus kepada ilmu Matematika dan bagi pihak yang membutuhkan.

*Wassalamu Alaikum Wr. Wb.*

Palopo, April 2020

Penulis,

Aldi Wahyudi

## RIWAYAT HIDUP



**Aldi Wahyudi**, lahir di Ketulungan pada tanggal 19 April 1998. Anak pertama dari dua bersaudara dari pasangan Bapak Supriadi dan Ibu Yuliati. Penulis mulai memasuki jenjang pendidikan dasar di SDN 169 Ketulungan, di Desa Ketulungan Kecamatan Sukamaju Kabupaten Luwu Utara pada tahun 2003 dan tamat pada tahun 2009. Pada tahun yang sama penulis melanjutkan pendidikan ke tingkat Sekolah Menengah Pertama yaitu SMP Negeri 4 Sukamaju di Desa Ketulungan Kecamatan Sukamaju Kabupaten Luwu Utara dan tamat pada tahun 2012. Kemudian pada tahun yang sama yaitu 2012 penulis melanjutkan ke tingkat atas yaitu SMK Negeri 1 Sukamaju (SMK Negeri 3 Luwu Utara) di Desa Minanga Tallu Kecamatan Sukamaju Kabupaten Luwu Utara dan tamat pada tahun 2015. Selanjutnya pada tahun 2015 penulis melanjutkan studinya dan terdaftar sebagai mahasiswa Program Studi Matematika Fakultas Sains (FSAINS) Universitas Cokroaminoto Palopo (UNCP). Selama mengikuti pendidikan di kampus, penulis aktif sebagai pengurus Himpunan Mahasiswa Program Studi (HMPS) Matematika, yang sekarang menjadi Himpunan Mahasiswa Matematika SAINS (HIMATIKA SAINS).

Pada tahun 2018 penulis pernah melaksanakan PKL (Praktek Kerja Lapangan) yang bertempat di PT. Asuransi Jiwasraya (Persero) Cabang Palopo, ditahun yang sama penulis juga pernah mengikuti KKN-PPM (Kuliah Kerja Nyata – Pembelajaran Pemberdayaan Masyarakat) yang bertempat di Kabupaten Pinrang, tepatnya di Kelurahan Langnga Kecamatan Mattiro Sompe Kabupaten Pinrang Provinsi Sulawesi Selatan.

## DAFTAR ISI

	Halaman
HALAMAN JUDUL .....	i
HALAMAN PENGESAHAN.....	ii
ABSTRAK .....	iii
KATA PENGANTAR .....	iv
RIWAYAT HIDUP.....	vi
DAFTAR ISI.....	vii
DAFTAR GAMBAR .....	viii
<b>BAB I     PENDAHULUAN</b>	
1.1   Latar Belakang.....	1
1.2   Rumusan Masalah.....	2
1.3   Tujuan Penelitian .....	2
1.4   Manfaat Penelitian .....	2
<b>BAB II    TINJAUAN PUSTAKA</b>	
2.1   Kajian Teori .....	3
2.2   Penelitian Yang Relevan.....	10
2.3   Kerangka Pikir .....	11
<b>BAB III   METODE PENELITIAN</b>	
3.1   Jenis dan Desain Penelitian.....	12
3.2   Lokasi dan Waktu Penelitian .....	12
3.3   Teknik Analisis Data .....	12
<b>BAB IV    HASIL DAN PEMBAHASAN</b>	
4.1   Jarak antar Lokasi Wisata .....	13
4.2   Penentuan Lintasan Terpendek.....	14
<b>BAB V     KESIMPULAN DAN SARAN</b>	
5.1   Kesimpulan .....	20
5.2   Saran .....	20
DAFTAR PUSTAKA .....	21



## DAFTAR GAMBAR

1 Jembatan Konisgberg .....	3
2 Representasi Masalah Jembatan.....	3
3 Graf Sederhana.....	4
4 Graf Tak Sederhana.....	4
5 Graf Berbobot.....	5
6 Graf Berarah.....	5
7 Flowchart Algoritma <i>Floyd Warshall</i> .....	11
8 Graf Jarak antar Lokasi Wisata .....	13

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Sektor pariwisata merupakan salah satu aset bangsa yang perlu dikembangkan. Dengan adanya potensi alam sebagai objek wisata dan juga minat warga lokal maupun mancanegara untuk berinteraksi dengan alam semakin besar. Suhamdani (2010) menegaskan bahwa pariwisata merupakan industri gaya baru yang mampu menyediakan pertumbuhan ekonomi yang cepat dalam hal kesempatan kerja, pendapatan, taraf hidup dan dalam mengaktifkan sektor lain di dalam negara penerima Wisatawan. Dengan demikian pariwisata tidak hanya menjadi upaya untuk pengelolaan alam namun juga dapat menjadi salah satu pengelola sistem keuangan.

Sebagaimana kota-kota lain di Sulawesi Selatan yang terkenal dengan objek wisatanya, Kota Palopo juga merupakan salah satu kota di Sulawesi Selatan yang memiliki banyak objek wisata yang dapat dikunjungi oleh para wisatawan. Dengan adanya banyak pilihan wisata, maka Wisatawan akan mempertimbangkan efisiensi waktu untuk menuju salah satu destinasi wisata. Dengan demikian metode lintasan terpendek (*Shortest Path*) dapat digunakan sebagai salah satu sarana untuk mengestimasi jarak terdekat untuk menuju ke destinasi wisata di Kota Palopo.

Masalah jalur terpendek adalah masalah menemukan suatu jalur antara dua simpul sedemikian sehingga jumlah bobot dari busur penyusunnya dapat seminimal mungkin (R. Kumar dan M. Kumar, 2010). Beberapa algoritma yang telah dikembangkan untuk menyelesaikan permasalahan jalur terpendek diantaranya algoritma *Dijkstra*, algoritma *Floyd Warshall* dan algoritma *Bellman-Ford*. Sementara metode yang paling efisien untuk permasalahan jalur terpendek dalam jaringan data adalah algoritma *Dijkstra*. Namun, pada jaringan dinamis yang sangat besar, algoritma *Dijkstra* menjadi tidak efisien karena simpul-simpul pada jaringan akan dikunjungi kembali sehingga banyak komputasi atau perhitungan-perhitungan yang diulang (R. Kumar dan M. Kumar, 2010).

Untuk menentukan jalur terpendek pada pencarian destinasi wisata di Kota Palopo dalam penelitian ini menggunakan metode *Floyd Warshall* untuk

melakukan perhitungan jalur terpendek. *Floyd Warshall* merupakan salah satu algoritma pencarian yang dapat digunakan dalam menghitung jalur terpendek, dan mampu membandingkan semua kemungkinan lintasan pada graf untuk setiap sisi dari semua simpul yang ada. Oleh sebab itu, algoritma ini cocok digunakan dalam menghadapi permasalahan perhitungan jalur terpendek khususnya dalam penerapannya pada pencarian lokasi destinasi wisata.

## **1.2 Rumusan Masalah**

Berdasarkan latar belakang tersebut, maka rumusan masalah pada penelitian ini adalah bagaimana implementasi algoritma *Floyd Warshall* pada optimasi lintasan terpendek untuk mencari destinasi wisata di Kota Palopo?

## **1.3 Tujuan Penelitian**

Adapun tujuan yang ingin dicapai dalam penelitian ini adalah untuk mengetahui implementasi algoritma *Floyd Warshall* pada optimasi lintasan terpendek untuk mencari destinasi wisata di Kota Palopo.

## **1.4 Manfaat Penelitian**

### 1) Bagi Peneliti

Manfaat penelitian ini bagi peneliti adalah untuk pengaplikasian ilmu matematika khususnya ilmu graf dalam kehidupan sehari-hari serta memberikan gambaran tentang implementasi algoritma *Floyd Warshall* dalam menentukan lintasan terpendek.

### 2) Bagi Pembaca

Penelitian ini diharapkan dapat menjadi salah satu sumber referensi dibidang pengetahuan matematika, khususnya graf. Penelitian ini diharapkan juga bisa dijadikan acuan dan literatur tambahan untuk penelitian selanjutnya yang memiliki relevansi dengan penelitian ini.

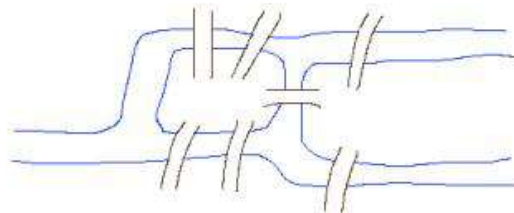
## BAB II

### TINJAUAN PUSTAKA

#### 2.1 Kajian Teori

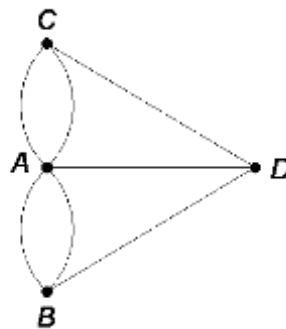
##### Teori Dasar Graf

(Syahputra, 2017) menyatakan bahwa graf terdiri atas himpunan simpul  $V$  dan himpunan sisi  $E$  dengan setiap sisi memiliki ujung-ujung yang berupa simpul. Graf dapat digunakan untuk mempresentasikan berbagai macam sistem nyata, dengan simpul menyatakan unsur dalam sistem tersebut dan unsur-unsur yang saling berhubungan digambarkan dengan adanya sisi yang menghubungkan unsur-unsur itu. Dalam sejarahnya, graf digunakan oleh Euler untuk memecahkan masalah jembatan Königsberg. Masalah jembatan Königsberg merupakan teka-teki yakni dari salah satu tempat tertentu apakah kita dapat berjalan dengan melalui ke tujuh jembatan itu masing-masing tepat satu kali.



Gambar 1. Jembatan Königsberg

Banyak orang sudah mencoba melakukannya dengan berbagai cara namun tidak ada yang berhasil. Hal ini menarik Leonard Euler untuk memecahkan masalah tersebut dengan menggunakan konsep yang sekarang dikenal sebagai teori graf. Jika setiap tempat diwakili oleh simpul dan setiap jembatan diwakili oleh sisi, maka masalah tersebut dapat digambarkan sebagai graf dengan empat simpul dan tujuh sisi (Syahputra, 2017).



Gambar 2. Representasi Masalah Jembatan

## Definisi Graf

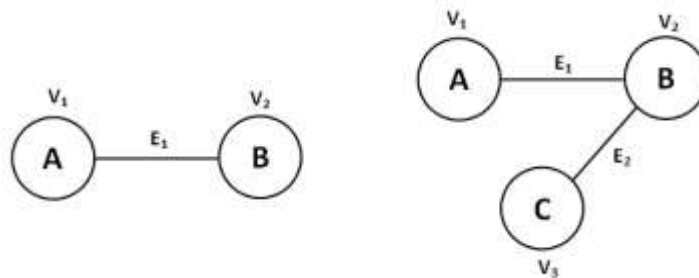
Menurut Mandailina (2009) Graf  $G$  adalah pasangan himpunan  $(V,E)$  dengan  $V$  adalah himpunan tidak kosong dan berhingga dari obyek-obyek yang disebut sebagai titik dan  $E$  adalah himpunan (mungkin kosong) pasangan tak berurutan dari titik-titik berbeda di  $V$  yang disebut sebagai sisi menurut Chartrand dan Lesniak, 1986:4 (dalam Mandailina 209).

## Jenis-jenis Graf

Menurut Zuhri (2016) berdasarkan ada tidaknya gelang atau sisi ganda pada suatu graf, graf dapat digolongkan menjadi 4 jenis:

### a) Graf Sederhana

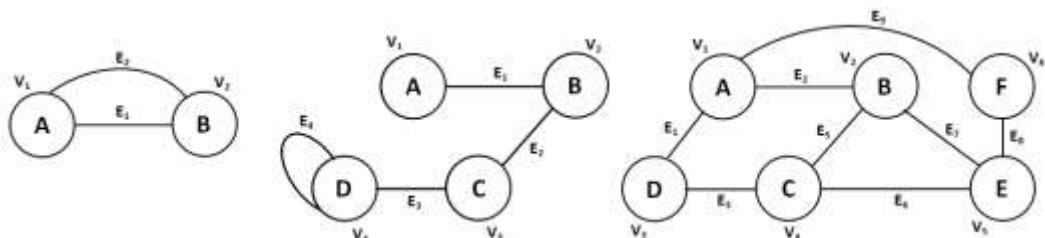
Graf sederhana adalah graf yang tidak memiliki garis paralel atau hanya satu simpul (*vertex*) saja yang memiliki garis paralel dan juga tidak memiliki gelang (*loop*). Garis paralel merupakan kondisi ketika satu simpul (*vertex*) memiliki lebih dari satu garis. Garis gelang adalah ketika ada garis yang berasal dari satu simpul dan kembali pada simpul tersebut.



Gambar 3. Graf sederhana

### b) Graf tak Sederhana

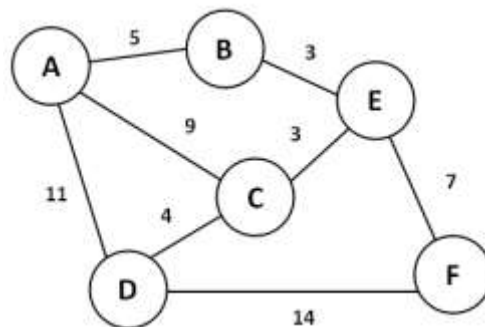
Graf tidak sederhana adalah graf yang memiliki garis paralel lebih dari satu simpul dan juga memiliki gelang (*loop*) disalah satu simpul.



Gambar 4. Graf tak sederhana

### c) Graf Berbobot

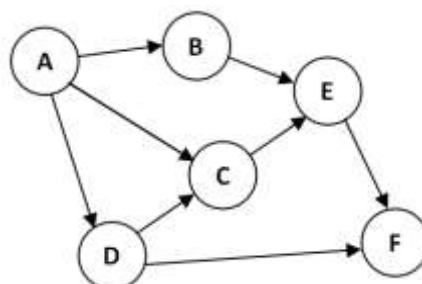
Graf berbobot adalah graf yang memiliki nilai pada setiap garisnya (*edge*). Nilai pada tiap *edge* dapat berbeda-beda bergantung pada masalah yang dimodelkan dengan graf. Aplikasinya biasa digunakan pada sistem rute lalu lintas dimana bobot pada rute lalu lintas merupakan nilai dari jarak antar dua tempat pada suatu kota. Jika pada graf menggambarkan sebuah peta jalan raya antar kota-kota maka sebuah bilangan pada *edge* adalah menunjukkan jarak antar kedua kota yang dihubungkan oleh *edge* tersebut.



Gambar 5. Graf berbobot

### d) Graf Berarah

Graf berarah adalah graf yang setiap sisinya memiliki orientasi arah dari suatu simpul ke simpul lainnya. Setiap *edge* menghubungkan pasangan verteks terurut, misalkan jika terdapat sebuah *edge*  $a$  yang menghubungkan pasangan verteks terurut  $(A,B)$  maka dapat ditulis dengan notasi  $a=(A,B)$ , yang menyatakan sebuah *edge* dari  $A$  ke  $B$ .



Gambar 6. Graf berarah

## Teori Dasar Lintasan Terpendek (*Shortest Path*)

Pemasalahan lintasan /rute terpendek yaitu menemukan lintasan terpendek antara dua atau beberapa simpul lebih yang saling berhubungan. Menurut Hayati dan Yohanes 2014 (dalam Syahputra, 2017) persoalan mencari lintasan terpendek didalam graf merupakan salah satu persoalan optimasi. Graf yang digunakan

dalam pencarian lintasan terpendek adalah graf berbobot (*weighted graph*), yaitu graf yang setiap sisinya diberikan suatu nilai atau bobot. Bobot pada sisi graf dapat menyatakan jarak antar kota, waktu pengiriman pesan, ongkos pembangunan, dan lain sebagainya. Asumsi yang digunakan di sini adalah bahwa semua bobot bernilai positif. Lintasan terpendek adalah jalur yang dilalui dari suatu *node* ke *node* lain dengan besar atau nilai pada sisi yang jumlah akhirnya dari *node* awal ke *node* akhir paling kecil. Lintasan terpendek adalah lintasan minimum yang diperlukan untuk mencapai suatu tempat dari tempat lain. Lintasan minimum yang dimaksud dapat dicari dengan menggunakan graf. Graf yang digunakan adalah graf yang berbobot yaitu graf setiap sisinya diberikan suatu nilai atau bobot.

Berikut contoh penerapan pencarian lintasan/rute terpendek :

- 1) Implementasi algoritma *Dijkstra* dalam aplikasi untuk menentukan lintasan terpendek jalan darat antar kota (Fitri dan Triansyah, 2013). Dalam penelitian tersebut disimpulkan perlu sebuah algoritma untuk dapat menyelesaikan persoalan rute terpendek yaitu dengan algoritma *Dijkstra*.
- 2) Implementasi algoritma *Dijkstra* pada kartu FPGA (*Field-Programmable Gate Array*) untuk perhitungan telkom (Benaicha dan Taibi, 2013). Jaringan merupakan satu set perangkat komputer yang digunakan untuk memberikan arus informasi, untuk rute informasi yang benar jaringan menggunakan proses routing dengan menggabungkan fleksibilitas dan kecepatan sehingga diperlukan teknologi baru yaitu FPGA.
- 3) Penemuan rute terpendek pada aplikasi berbasis peta (Wira Buana, 2010). Pada penelitian ini menghasilkan rute terpendek untuk berbagai keperluan masyarakat yang dihadapkan dengan permasalahan transportasi seperti kemacetan pada jalan raya.

### **Algoritma Rute Terpendek (*Shortest Path Algorithm*)**

Algoritma yang dapat digunakan untuk mencari rute terpendek telah banyak diteliti. Beberapa algoritma yang dapat digunakan untuk penyelesaian penentuan rute terpendek Sanan, dkk, 2013 (dalam Syahputra 2019), yaitu :

1) Algoritma *Dijkstra*

*Dijkstra* adalah algoritma berdasarkan *Greedy* dan memecahkan satu sumber masalah jalan terpendek.

2) Algoritma *Bellman-Ford*

*Bellman-Ford* adalah algoritma berbasis Pemrograman Dinamis.

3) Algoritma *A\* Search*

Algoritma *A\* Search* adalah algoritma pencarian graf/pohon yang menemukan jalur dari *node* awal yang diberikan ke node tujuan tertentu.

4) Algoritma *Floyd Warshall*

Algoritma ini memecahkan semua pasangan jalur terpendek dalam grafik tepi arah.

5) Algoritma *Johnson*

Algoritma ini memecahkan semua pasangan jalur terpendek, di jarang tertimbang, grafik berarah.

**Algoritma *Floyd Warshall***

Algoritma *Floyd Warshall* diperkenalkan oleh Robert Floyd pada tahun 1962. Namun, pada dasarnya memiliki kesamaan dengan algoritma yang pernah diperkenalkan sebelumnya oleh Bernard Roy pada tahun 1959 dan juga Stephen Warshall pada 1962. Algoritma *Floyd Warshall* memiliki input graf berarah dan berbobot  $(V,E)$ , yang berupa daftar titik (node/vertex  $V$ ) dan daftar sisi (edge  $E$ ). Jumlah bobot sisi-sisi pada sebuah jalur adalah bobot jalur tersebut. Sisi pada  $E$  diperbolehkan memiliki bobot negatif, akan tetapi tidak diperbolehkan memiliki siklus dengan bobot negatif. Algoritma ini menghitung bobot terkecil dari semua jalur yang menghubungkan sebuah pasangan titik, dan melakukannya sekaligus untuk semua pasangan titik.

Algoritma *Floyd Warshall* merupakan pemrograman dinamis yang melakukan penyelesaian masalah dengan melihat solusi yang akan diperoleh sebagai suatu keputusan yang saling terkait. Prinsip optimalitas ialah prinsip yang diterapkan oleh pemrograman dinamis, yaitu jika solusi total optimal, maka bagian solusi sampai suatu tahap juga optimal. Algoritma *FloydWarshall* memiliki input graf berarah dan berbobot serta output dari algoritma *Floyd Warshall* adalah dengan menghitung bobot terkecil dari semua lintasan yang



menghubungkan sebuah pasangan titik dan melakukannya sekaligus untuk semua pasangan titik.

Algoritma *Floyd Warshall* untuk mencari path terpendek merupakan algoritma yang sederhana dan mudah implementasinya. Prinsip dasarnya tidak jauh berbeda dengan algoritma *Dijkstra*. Pada iterasi ke-1, dihitung jarak terpendek dari semua titik ke semua titik apabila pathnya melalui titik  $v_i$ . Misalkan  $W^{(0)}$  adalah matriks hubung graf berarah berlabel mula-mula.  $W^*$  adalah matriks hubung minimal dengan  $W_{ij}^*$  = path terpendek dari titik  $v_i$  ke  $v_j$  (Siang J.J, 2014).

Algoritma *Floyd Warshall* untuk mencari path terpendek adalah sebagai berikut:

- 1)  $W = W^{(0)}$
- 2) Untuk  $k = 1$  hingga  $n$ , lakukan: {pengujian jarak lewat titik  $v_k$ }
  - Untuk  $i = 1$  hingga  $n$ , lakukan:
  - Untuk  $k = 1$  hingga  $n$ , lakukan:
  - Jika  $W_{ij} > W_{i,k} + W_{k,j}$  maka tukar  $W_{ij}$  dengan  $W_{i,k} + W_{k,j}$
- 3)  $W^* = W$

Meskipun waktu prosesnya bukanlah yang tercepat, algoritma *Floyd Warshall* sering dipergunakan untuk menghitung path terpendek karena kesederhanaannya. Disamping itu, program implementasi algoritma *Floyd Warshall* sangat mudah dibuat.

Algoritma *Floyd Warshall* diatas hanya menghitung jarak terpendek dari semua titik ke semua titik, tetapi tidak menjelaskan bagaimana path terpendeknya. Untuk menentukan path yang menghasilkan jarak terpendek, tambahkan matriks bujur sangkar (ukuran  $n \times n$ )  $Z$  yang disusun sebagai berikut:

Inisialisasi:

$$Z^{(0)}_{i,j} = \begin{cases} j & \text{jika } W^{(0)}_{i,j} \neq \infty \\ 0 & \text{jika } W^{(0)}_{i,j} = \infty \end{cases} \dots\dots\dots(i)$$

Dalam iterasi ke- $k$ , apabila titik  $v_k$  disisipkan antara titik- $i$  dan titik- $j$  (berarti menukan  $W_{i,j} + W_{k,j}$ ), maka ganti  $Z_{ij}$  dengan  $Z_{ik}$ . Agar lebih efisien penggantian matriks  $Z$  dilakukan bersama-sama dengan iterasi pencarian jarak

terpendeknya. Jarak terpendek dari titik  $v_i$  ke titik  $v_j$  dilakukan melalui titik  $v_1, v_2, \dots, v_q$  yang didapat dari  $Z$  optimalnya sebagai berikut:

$$v_1 = Z_{ij}$$

$$v_2 = Z_{v_1, j}$$

$$v_3 = Z_{v_2, j}$$

....

$$v_j = Z_{v_q, j}$$

Revisi algoritma *Floyd Warshall* dengan melibatkan path terpendeknya adalah sebagai berikut:

- 1)  $W = W^{(0)} ; Z = Z^{(0)}$
- 2) Untuk  $k=1$  hingga  $n$ , lakukan: {pengujian lewat titik  $v_k$ }
  - Untuk  $i=1$  hingga  $n$ , lakukan:
  - Untuk  $j=1$  hingga  $n$ , lakukan:
    - Jika  $W_{i,j} > W_{i,k} + W_{k,j}$  maka
    - a. Tukar  $W_{i,j}$  dengan  $W_{i,k} + W_{k,j}$
    - b. Ganti  $Z_{ij}$  dengan  $Z_{ik}$
- 3)  $W = W^{(7)}$

### **Objek Wisata di Kota Palopo**

Pengertian objek wisata dalam UU No. 9 Tahun 1990 Tentang Kepariwisata Bab I Pasal 1.6 menyebutkan objek wisata dan daya tarik wisata adalah segala sesuatu yang menjadi sasaran wisata. Selanjutnya dalam Bab III Pasal 4 disebutkan bahwa objek wisata dan daya tarik wisata terdiri dari atas (Puspita, 2012):

- a) Objek dan daya tarik wisata ciptaan Tuhan Yang Maha Esa yang berwujud keadaan flora dan fauna.
- b) Objek wisata dan daya tarik wisata hasil karya manusia yang berwujud museum, peninggalan purbakala, peninggalan sejarah, seni budaya, wisata agro, wisata tirta, wisata buru, wisata petualang, tempat rekreasi dan tempat hiburan.

Pariwisata di Kota Palopo dikenal memiliki beberapa objek wisata, diantaranya Permandian Umum Rampoang, Wisata Air Terjun Latuppa, Bukit Kambo, Permandian Waterboom Agrowisata, Lapangan Pancasila, Pelabuhan Tanjung Ringgit, dan Pantai Labombo.

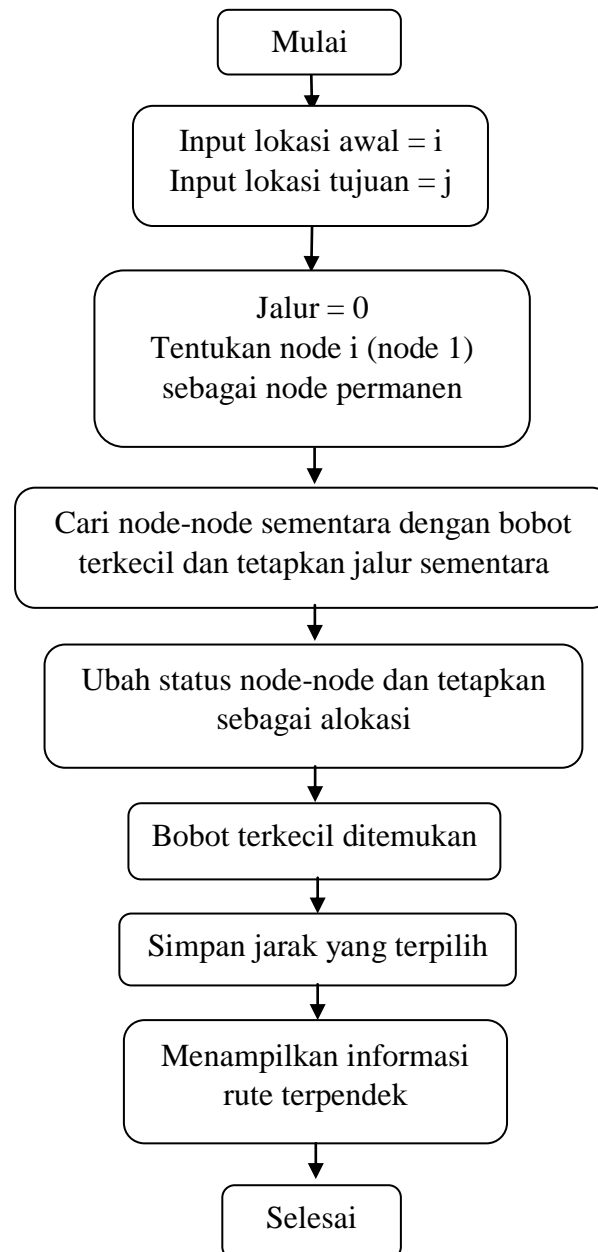
## 2.2 Penelitian yang Relevan

Penelitian mengenai pencarian jalur terpendek menggunakan algoritma *Floyd Warshall* pernah dilakukan peneliti-peneliti sebelumnya dengan berbagai permasalahan yang berbeda. Andriani (2014) dalam penelitiannya membahas tentang rancang bangun sistem informasi rute wisata terpendek berbasis algoritma *Floyd Warshall*. Metode pengumpulan data dalam penelitian ini yaitu metode observasi untuk mempelajari jalur wisata Yogyakarta yang diperoleh dari *Jogja tourism maps* dan *google maps*. Peta wisata ditransformasikan ke dalam bentuk diagram grafik. Algoritma *Floyd Warshall* diterapkan dalam perhitungan bobot *path* dari diagram grafik dari peta wisata Yogyakarta untuk mencari rute terpendek ke beberapa obyek wisata populer di Yogyakarta yang hasilnya digunakan untuk membangun sistem informasi rute wisata terpendek menuju obyek wisata di Yogyakarta. Pembangunan sistem informasi berbentuk *website* sehingga mudah diakses secara *online*.

Novandi (2007) dalam penelitiannya membahas tentang perbandingan algoritma Dijkstra dan algoritma *Floyd Warshall* dalam penentuan lintasan terpendek. Berbagai kalangan menemui permasalahan serupa dengan variasi yang berbeda, contohnya seorang pengemudi yang mencari jalur terpendek dari tempat asal ke tempat tujuan dan seorang desainer jaringan komputer yang harus mendesain skema perutean pada jaringan yang dia tangani agar memaksimalkan performa jaringan dan meminimalkan beban yang harus ditangani oleh jaringan tersebut. Dalam penelitian ini membandingkan suatu graf berbobot yang merepresentasikan kondisi keterhubungan antarkota di suatu daerah yang sama dengan menerapkan algoritma *Dijkstra* dan algoritma *Floyd Warshall*. Hasil penelitian ini didapatkan bahwa algoritma *Dijkstra* yang menerapkan prinsip *Greedy* tidak selalu berhasil memberikan solusi optimum untuk kasus penentuan lintasan terpendek (*single pair shortest path*) dibandingkan dengan algoritma

*Floyd Warshall* yang menerapkan pemrograman dinamis lebih menjamin keberhasilan penemuan solusi optimum untuk kasus penentuan lintasan terpendek (*single pair shortest path*).

### 2.3 Kerangka Pikir



Gambar 7. Flowchart Algoritma *Floyd Warshall*

## **BAB III**

### **METODE PENELITIAN**

#### **3.1 Jenis dan Desain Penelitian**

Pada penelitian ini, penulis menggunakan metode penelitian kuantitatif dimana data yang digunakan data berbentuk angka. Data yang digunakan dalam penelitian ini adalah data sekunder. Data jarak objek wisata didapat dari *google maps*. Selain itu, data sekunder juga diperoleh dari berbagai literatur seperti buku teks, internet, jurnal online, dan sumber kepustakaan yang lainnya.

#### **3.2 Lokasi dan Waktu Penelitian**

Data penelitian ini dikumpulkan dari data jarak antar lokasi yang ada di *google maps*. Penelitian ini dilaksanakan selama bulan Desember 2019 sampai dengan Januari 2020.

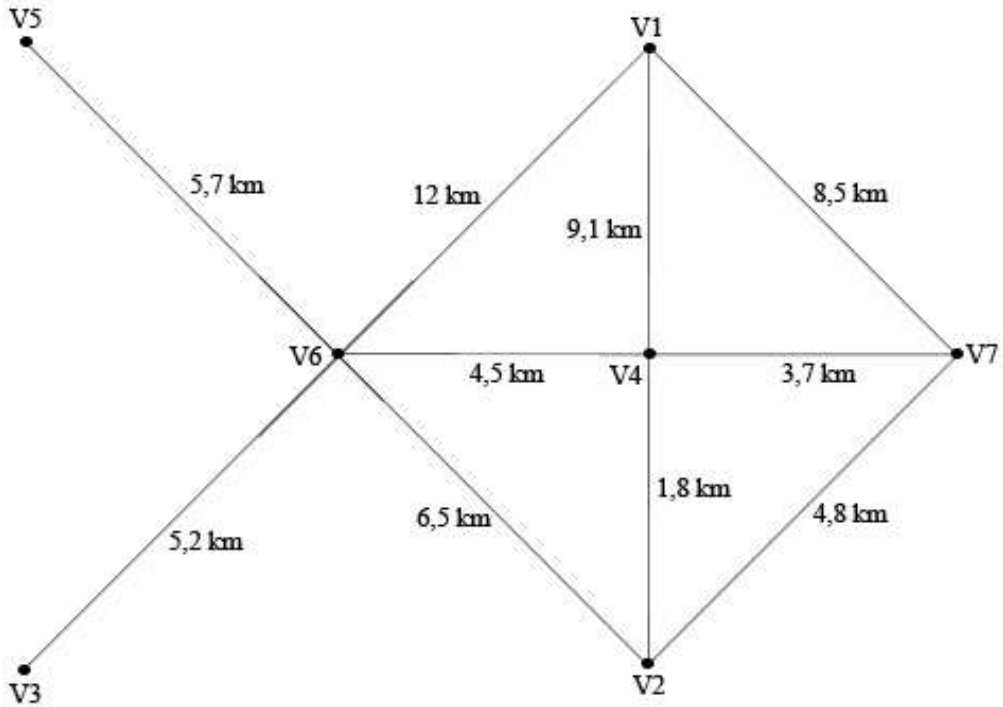
#### **3.3 Teknik Analisis Data**

Data yang diperoleh akan dianalisis secara kuantitatif. Data yang berhasil dikumpulkan secara kuantitatif terlebih dahulu diolah dan ditabulasikan. Data kuantitatif disajikan dalam bentuk tabel jarak antar lokasi obyek wisata dalam satuan meter. Data akan diproses dengan menggunakan perhitungan manual dengan mengikuti tahapan algoritma *Floyd Warshall*. Pencarian jalur terpendek tidak melihat atau menghiraukan suatu kondisi seperti jalur atau jalan yang sedang rusak, kepadatan dan rambu lalu lintas sehingga terjadi pengalihan jalan ke jalur lainnya, semua jalur dianggap dalam keadaan normal.

**BAB IV**  
**HASIL DAN PEMBAHASAN**

**4.1 Jarak antar Lokasi Wisata**

Berikut adalah graf jarak antar lokasi wisata di Kota Palopo:



Gambar 8. Graf jarak antar lokasi wisata

Gambar 8 dapat direpresentasikan dalam bentuk tabel sebagai berikut.

Tabel Jarak antar lokasi wisata

	V <sub>1</sub>	V <sub>2</sub>	V <sub>3</sub>	V <sub>4</sub>	V <sub>5</sub>	V <sub>6</sub>	V <sub>7</sub>
v <sub>1</sub>	∞	∞	∞	9,1 km	∞	12 km	8,5 km
v <sub>2</sub>	∞	∞	∞	1,8	∞	6,5 km	4,8 km
v <sub>3</sub>	∞	∞	∞	∞	∞	5,2 km	∞
v <sub>4</sub>	9,1 km	1,8 km	∞	∞	∞	4,9 km	3,7 km
v <sub>5</sub>	∞	∞	∞	∞	∞	5,7 km	∞
v <sub>6</sub>	12 km	6,5 km	5,2 km	4,9 km	5,7 km	∞	∞
v <sub>7</sub>	8,5 km	4,8 km	∞	3,7 km	∞	∞	∞

Sumber: Data sekunder sebelum diolah (2019)

Keterangan:

v<sub>1</sub> = Permandian Umum Rampoang

v<sub>2</sub> = Pantai Labombo

$v_3$  = Wisata Air Terjun Latuppa

$v_4$  = Lapangan Pancasila

$v_5$  = Bukit Kambo

$v_6$  = Agrowisata Waterboom

$v_7$  = Pelabuhan Tanjung Ringgit

#### 4.2 Penentuan Lintasan Terpendek

Dari data tersebut diketahui jarak dari  $v_1$  ke  $v_4$  adalah 9,1 km,  $v_1$  ke  $v_6$  adalah 12 km,  $v_1$  ke  $v_7$  adalah 8,5 km dan seterusnya. Kemudian tanda “ $\infty$ ” pada tabel tersebut menandakan bahwa tidak adanya path langsung dari titik asal ke titik tujuan, setelah itu permasalahan ini akan direpresentasikan ke dalam matriks  $n \times n$ , kemudian dicari lintasan terpendek (*Shortest Path*) menggunakan algoritma *Floyd Warshall*. Untuk menentukan rute terpendek dari 7 destinasi wisata di Kota Palopo, terlebih dahulu dilakukan perhitungan menggunakan matriks  $n \times n$ .

$$W = W^{(0)} = \begin{matrix} & \begin{matrix} v_1 & v_2 & v_3 & v_4 & v_5 & v_6 & v_7 \end{matrix} \\ \begin{matrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \end{matrix} & \begin{bmatrix} \infty & \infty & \infty & 9,1 & \infty & 12 & 8,5 \\ \infty & \infty & \infty & 1,8 & \infty & 6,5 & 4,8 \\ \infty & \infty & \infty & \infty & \infty & 5,2 & \infty \\ 9,1 & 1,8 & \infty & \infty & \infty & 4,9 & 3,7 \\ \infty & \infty & \infty & \infty & \infty & 5,7 & \infty \\ 12 & 6,5 & 5,2 & 4,9 & 5,7 & \infty & \infty \\ 8,5 & 4,8 & \infty & 3,7 & \infty & \infty & \infty \end{bmatrix} \end{matrix}$$

Matriks path  $Z^{(0)}$  dibuat menurut dengan persamaan (i)

Ini berarti bahwa jika titik  $v_i$  dihubungkan langsung dengan titik  $v_j$ , maka elemen matriks  $Z_{i,j}$  sama dengan nilai kolom ( $=j$ )

$$\text{Diperoleh } Z^{(0)} = \begin{matrix} & \begin{matrix} v_1 & v_2 & v_3 & v_4 & v_5 & v_6 & v_7 \end{matrix} \\ \begin{matrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \end{matrix} & \begin{bmatrix} 0 & 0 & 0 & 4 & 0 & 6 & 7 \\ 0 & 0 & 0 & 4 & 0 & 6 & 7 \\ 0 & 0 & 0 & 0 & 0 & 6 & 0 \\ 1 & 2 & 0 & 0 & 0 & 6 & 7 \\ 0 & 0 & 0 & 0 & 0 & 6 & 0 \\ 1 & 2 & 3 & 4 & 5 & 0 & 0 \\ 1 & 2 & 0 & 4 & 0 & 0 & 0 \end{bmatrix} \end{matrix}$$

Iterasi untuk  $k = 1$

Untuk setiap sel matriks  $W$  dicek apakah  $W[i,j] > W[i,k] + W[k,j]$ . Jika ya, maka  $W[i,j]$  diganti dengan  $W[i,k] + W[k,j]$ . Sebagai contoh:

- a.  $W[1,6] = 12$ , sedangkan  $W[1,1] + W[1,6] = \infty + 12 = \infty$ . Karena  $W[1,2]$  tidak  $>$   $W[1,1] + W[1,2]$ , maka  $W[1,2]$  tidak diubah.

$$W^{(1)} = \begin{matrix} & v_1 & v_2 & v_3 & v_4 & v_5 & v_6 & v_7 \\ \begin{matrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \end{matrix} & \left[ \begin{array}{ccccccc} \infty & \infty & \infty & 9,1 & \infty & 12 & 8,5 \\ \infty & \infty & \infty & 1,8 & \infty & 6,5 & 4,8 \\ \infty & \infty & \infty & \infty & \infty & 5,2 & \infty \\ 9,1 & 1,8 & \infty & \infty & \infty & 4,9 & 3,7 \\ \infty & \infty & \infty & \infty & \infty & 5,7 & \infty \\ 12 & 6,5 & 5,2 & 4,9 & 5,7 & \infty & \infty \\ 8,5 & 4,8 & \infty & 3,7 & \infty & \infty & \infty \end{array} \right. \end{matrix}$$

Untuk iterasi  $k = 1$  matriks  $W^{(1)}$  selnya tidak ada yang berubah karena tidak ada nilai bobot yang berubah, begitu juga dengan matriks  $Z^{(1)}$  dimana tidak ada sel yang berubah. Jadi matriks  $W^{(1)}$  sama nilainya dengan matriks  $W^{(0)}$  dan matriks  $Z^{(1)}$  sama nilainya dengan matriks  $Z^{(0)}$ .

$$Z^{(1)} = \begin{matrix} & v_1 & v_2 & v_3 & v_4 & v_5 & v_6 & v_7 \\ \begin{matrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \end{matrix} & \left[ \begin{array}{ccccccc} 0 & 0 & 0 & 4 & 0 & 6 & 7 \\ 0 & 0 & 0 & 4 & 0 & 6 & 7 \\ 0 & 0 & 0 & 0 & 0 & 6 & 0 \\ 1 & 2 & 0 & 0 & 0 & 6 & 7 \\ 0 & 0 & 0 & 0 & 0 & 6 & 0 \\ 1 & 2 & 3 & 4 & 5 & 0 & 0 \\ 1 & 2 & 0 & 4 & 0 & 0 & 0 \end{array} \right. \end{matrix}$$

Iterasi untuk  $k = 2$

Iterasi untuk  $k=2$  dilakukan dengan cara yang sama seperti iterasi untuk  $k = 1$ , hanya titik perantaranya adalah titik  $v_2$ . Sebagai contoh:

- a)  $W[6,7] = \infty$ , sedangkan  $W[6,2] + W[2,7] = 6,5+4,8 = 11,3$ . Karena  $W[6,7] >$   $W[6,2] + W[2,7]$  maka harga  $W[6,7]$  diganti dengan 11,3 Ini berarti bahwa lintasan dari  $v_6$  ke  $v_7$  melalui  $v_2$  ( $v_6v_2v_7$ ) lebih pendek dibandingkan lintasan  $v_6$  ke  $v_7$  secara langsung atau melalui  $v_2$  atau sebaliknya.



Dengan cara yang sama, harga  $W_{ij}$  dihitung untuk setiap  $i$  dan  $j$ . didapatkan matriks  $W^{(2)}$  yang ditebalkan menunjukkan nilai yang berubah dibandingkan dengan nilai matriks  $W^{(1)}$  dan  $W^{(0)}$ . Ini berarti bahwa perjalanan dari titik  $v_i$  ke  $v_j$  pada elemen yang ditebalkan akan lebih optimal jika dilakukan melalui titik  $v_2$ . Sel dengan nilai  $W^{(2)}$  yang berubah akan menghasilkan perubahan pula pada matriks path  $Z^{(2)}$ .

$Z^{(2)}_{i,j} = Z^{(1)}_{i,k} = Z^{(1)}_{i,1}$ . Maka:

$$Z^{(2)}_{4,3} = Z^{(1)}_{4,2} = 2. \quad Z^{(2)}_{6,5} = Z^{(1)}_{6,2} = 2. \quad Z^{(2)}_{6,6} = Z^{(1)}_{6,2} = 2. \quad Z^{(2)}_{7,6} = Z^{(1)}_{7,2} = 2. \quad Z^{(2)}_{7,7} = Z^{(1)}_{7,2} = 2$$

Proses yang sama dilakukan untuk semua titik  $i$  dan  $j$ , didapatkan :

$$W^{(2)} = \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \end{array} \begin{bmatrix} v_1 & v_2 & v_3 & v_4 & v_5 & v_6 & v_7 \\ \infty & \infty & \infty & 9,1 & \infty & 12 & 8,5 \\ \infty & \infty & \infty & 1,8 & \infty & 6,5 & 4,8 \\ \infty & \infty & \infty & \infty & \infty & 5,2 & \infty \\ 9,1 & 1,8 & \infty & \mathbf{3,6} & \infty & 4,9 & 3,7 \\ \infty & \infty & \infty & \infty & \infty & 5,7 & \infty \\ 12 & 6,5 & 5,2 & 4,9 & 5,7 & \mathbf{13} & \mathbf{11,3} \\ 8,5 & 4,8 & \infty & 3,7 & \infty & \mathbf{11,3} & \mathbf{9,6} \end{bmatrix}$$

$$Z^{(2)} = \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \end{array} \begin{bmatrix} v_1 & v_2 & v_3 & v_4 & v_5 & v_6 & v_7 \\ 0 & 0 & 0 & 4 & 0 & 6 & 7 \\ 0 & 0 & 0 & 4 & 0 & 6 & 7 \\ 0 & 0 & 0 & 0 & 0 & 6 & 0 \\ 1 & 2 & 0 & \mathbf{2} & 0 & 6 & 7 \\ 0 & 0 & 0 & 0 & 0 & 6 & 0 \\ 1 & 2 & 3 & 4 & 5 & \mathbf{2} & \mathbf{2} \\ 1 & 2 & 0 & 4 & 0 & \mathbf{2} & \mathbf{2} \end{bmatrix}$$

Dengan cara yang sama, untuk iterasi  $k = 3, 4, 5, 6, 7$ , diperoleh matriks:

$$W^{(3)} = \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \end{array} \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \end{array} \begin{bmatrix} \infty & \infty & \infty & 9,1 & \infty & 12 & 8,5 \\ \infty & \infty & \infty & 1,8 & \infty & 6,5 & 4,8 \\ \infty & \infty & \infty & \infty & \infty & 5,2 & \infty \\ 9,1 & 1,8 & \infty & 3,6 & \infty & 4,9 & 3,7 \\ \infty & \infty & \infty & \infty & \infty & 5,7 & \infty \\ 12 & 6,5 & 5,2 & 4,9 & 5,7 & \mathbf{10,4} & 11,3 \\ 8,5 & 4,8 & \infty & 3,7 & \infty & 11,3 & 9,6 \end{bmatrix}$$

$$Z^{(3)} = \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \end{array} \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \end{array} \begin{bmatrix} 0 & 0 & 0 & 4 & 0 & 6 & 7 \\ 0 & 0 & 0 & 4 & 0 & 6 & 7 \\ 0 & 0 & 0 & 0 & 0 & 6 & 0 \\ 1 & 2 & 0 & 2 & 0 & 6 & 7 \\ 0 & 0 & 0 & 0 & 0 & 6 & 0 \\ 1 & 2 & 3 & 4 & 5 & \mathbf{3} & 2 \\ 1 & 2 & 0 & 4 & 0 & 2 & 2 \end{bmatrix}$$

$$W^{(4)} = \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \end{array} \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \end{array} \begin{bmatrix} \mathbf{18,2} & \mathbf{10,9} & \infty & 9,1 & \infty & 12 & 8,5 \\ \mathbf{10,9} & \mathbf{3,6} & \infty & 1,8 & \infty & 6,5 & 4,8 \\ \infty & \infty & \infty & \infty & \infty & 5,2 & \infty \\ 9,1 & 1,8 & \infty & 3,6 & \infty & 4,9 & 3,7 \\ \infty & \infty & \infty & \infty & \infty & 5,7 & \infty \\ 12 & 6,5 & 5,2 & 4,9 & 5,7 & \mathbf{9,8} & \mathbf{8,6} \\ 8,5 & 4,8 & \infty & 3,7 & \infty & \mathbf{8,6} & \mathbf{7,4} \end{bmatrix}$$

$$Z^{(4)} = \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \end{array} \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \end{array} \begin{bmatrix} \mathbf{4} & \mathbf{4} & 0 & 4 & 0 & 6 & 7 \\ \mathbf{4} & \mathbf{4} & 0 & 4 & 0 & 6 & 7 \\ 0 & 0 & 0 & 0 & 0 & 6 & 0 \\ 1 & 2 & 0 & 2 & 0 & 6 & 7 \\ 0 & 0 & 0 & 0 & 0 & 6 & 0 \\ 1 & 2 & 3 & 4 & 5 & \mathbf{4} & \mathbf{4} \\ 1 & 2 & 0 & 4 & 0 & \mathbf{4} & \mathbf{4} \end{bmatrix}$$

$$W^{(5)} = \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \end{array} \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \end{array} \begin{bmatrix} 18,2 & 10,9 & \infty & 9,1 & \infty & 12 & 8,5 \\ 10,9 & 3,6 & \infty & 1,8 & \infty & 6,5 & 4,8 \\ \infty & \infty & \infty & \infty & \infty & 5,2 & \infty \\ 9,1 & 1,8 & \infty & 3,6 & \infty & 4,9 & 3,7 \\ \infty & \infty & \infty & \infty & \infty & 5,7 & \infty \\ 12 & 6,5 & 5,2 & 4,9 & 5,7 & 9,8 & 8,6 \\ 8,5 & 4,8 & \infty & 3,7 & \infty & 8,6 & 7,4 \end{bmatrix}$$

$$Z^{(5)} = \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \end{array} \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \end{array} \begin{bmatrix} 4 & 4 & 0 & 4 & 0 & 6 & 7 \\ 4 & 4 & 0 & 4 & 0 & 6 & 7 \\ 0 & 0 & 0 & 0 & 0 & 6 & 0 \\ 1 & 2 & 0 & 2 & 0 & 6 & 7 \\ 0 & 0 & 0 & 0 & 0 & 6 & 0 \\ 1 & 2 & 3 & 4 & 5 & 4 & 4 \\ 1 & 2 & 0 & 4 & 0 & 4 & 4 \end{bmatrix}$$

$$W^{(6)} = \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \end{array} \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \end{array} \begin{bmatrix} 18,2 & 10,9 & \mathbf{17,2} & 9,1 & \mathbf{17,7} & 12 & 8,5 \\ 10,9 & 3,6 & \mathbf{11,7} & 1,8 & \mathbf{12,2} & 6,5 & 4,8 \\ \mathbf{17,2} & \mathbf{11,7} & \mathbf{10,4} & \mathbf{9,8} & \mathbf{10,9} & 5,2 & \mathbf{13,8} \\ 9,1 & 1,8 & \mathbf{10,1} & 3,6 & \mathbf{10,6} & 4,9 & 3,7 \\ \mathbf{17,7} & \mathbf{12,2} & \mathbf{10,9} & \mathbf{10,3} & \mathbf{11,4} & 5,7 & \mathbf{14,3} \\ 12 & 6,5 & 5,2 & 4,9 & 5,7 & 9,8 & 8,6 \\ 8,5 & 4,8 & \mathbf{13,8} & 3,7 & \mathbf{14,3} & 8,6 & 7,4 \end{bmatrix}$$

$$Z^{(6)} = \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \end{array} \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \end{array} \begin{bmatrix} 4 & 4 & \mathbf{6} & 4 & \mathbf{6} & 6 & 7 \\ 4 & 4 & \mathbf{6} & 4 & \mathbf{6} & 6 & 7 \\ \mathbf{6} & \mathbf{6} & \mathbf{6} & \mathbf{6} & \mathbf{6} & 6 & \mathbf{6} \\ 1 & 2 & \mathbf{6} & 2 & \mathbf{6} & 6 & 7 \\ \mathbf{6} & \mathbf{6} & \mathbf{6} & \mathbf{6} & \mathbf{6} & 6 & \mathbf{6} \\ 1 & 2 & 3 & 4 & 5 & 4 & 4 \\ 1 & 2 & \mathbf{6} & 4 & \mathbf{6} & 4 & 4 \end{bmatrix}$$

$$\begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \end{array}$$

$$W^{(7)} = \begin{matrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \end{matrix} \begin{bmatrix} \mathbf{17} & 10,9 & 17,2 & 9,1 & 17,7 & 12 & 8,5 \\ 10,9 & 3,6 & 11,7 & 1,8 & 12,2 & 6,5 & 4,8 \\ 17,2 & 11,7 & 10,4 & 9,8 & 10,9 & 5,2 & 13,8 \\ 9,1 & 1,8 & 10,1 & 3,6 & 10,6 & 4,9 & 3,7 \\ 17,7 & 12,2 & 10,9 & 10,3 & 11,4 & 5,7 & 14,3 \\ 12 & 6,5 & 5,2 & 4,9 & 5,7 & 9,8 & 8,6 \\ 8,5 & 4,8 & 13,8 & 3,7 & 14,3 & 8,6 & 7,4 \end{bmatrix}$$

$$Z^{(7)} = \begin{matrix} & v_1 & v_2 & v_3 & v_4 & v_5 & v_6 & v_7 \\ v_1 & \mathbf{7} & 4 & 6 & 4 & 6 & 6 & 7 \\ v_2 & 4 & 4 & 6 & 4 & 6 & 6 & 7 \\ v_3 & 6 & 6 & 6 & 6 & 6 & 6 & 6 \\ v_4 & 1 & 2 & 6 & 2 & 6 & 6 & 7 \\ v_5 & 6 & 6 & 6 & 6 & 6 & 6 & 6 \\ v_6 & 1 & 2 & 3 & 4 & 5 & 4 & 4 \\ v_7 & 1 & 2 & 6 & 4 & 6 & 4 & 4 \end{bmatrix}$$

$W^*$  merupakan matriks jarak terpendek yang melalui titik  $v^*$  dan  $Z^*$  merupakan path yang harus dilalui, sebagai contoh  $W^{(7)}$  merupakan matriks jarak terpendek yang melalui  $v_7$  dan  $Z^{(7)}$  merupakan path yang harus dilalui. Dari hasil iterasi diperoleh salah satu contoh jarak terpendek yaitu dari  $v_5$  ke  $v_7$  adalah 14,3 km dan path yang harus dilalui adalah:

$$Z^{(7)}_{5,7} = 6$$

$$Z^{(7)}_{6,7} = 4$$

$$Z^{(7)}_{4,7} = 7$$

Jadi path terpendek dari  $v_5$  ke  $v_7$  adalah  $v_5 \rightarrow v_6 \rightarrow v_4 \rightarrow v_7$  dengan jarak terpendek 14,3 km.

## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **5.1 Kesimpulan**

Dari hasil penelitian dapat disimpulkan bahwa dengan menggunakan algoritma *Floyd Warshall* diperoleh lintasan terpendek dari semua titik ke semua titik pada destinasi wisata di Kota Palopo. Sebagai contoh, jarak terpendek titik  $v_5$  ke  $v_7$  adalah 14,3 km dengan lintasan  $v_5 \rightarrow v_6 \rightarrow v_4 \rightarrow v_7$ .

#### **5.2 Saran**

Diharapkan ada penelitian yang lebih lanjut untuk mengetahui efisiensi dari pencarian jalur terpendek menggunakan metode heuristik dan juga diharapkan penelitian berikutnya dapat membandingkan antar metode-metode heuristik yang lain.

## DAFTAR PUSTAKA

- Andriani, Anik. 2014. Rancang Bangun Sistem Informasi Rute Wisata Terpendek Berbasis Algoritma *Floyd Warshall*. *Bianglala Informatika Vol II*, No.1
- Benaicha, R. dan Taibi, M. 2013. *Dijkstra Algorithm Implementation On FPGA Card For Telkom Calculations*. *International Journal of Engineering Sciences & Emerging Technologies*, Volume 4, Issue 2, pp: 110:116©IJESET.
- Fitria dan Apri, T. 2013. Impelemntasi Algoritma Djikstra dalam Aplikasi untuk Menentukan Lintasan Terpendek Jalan Darat Antar Kota di Sumatera Bagian Selatan, *Jurnal Sistem Informasi(JSI)*, Vol. 5, No. 2.
- Mandailina, Vera. 2009. *Faktorisasi pada Graf Komplit*. Skripsi tidak diterbitkan. Malang: Fakultas Sains dan Teknologi-UIN Malang.
- Novandi, Raden. 2007. *Perbandingan Algoritma Dijkstra dan Algoritma Floyd-Warshall dalam Penentuan Lintasan Terpendek*. Makalah IF2251 Strategi Algoritmik. Bandung : STEI Institut Teknologi Bandung.
- Puspita, M. 2012. Undang-Undang Republik Indonesia No. 9 Tahun 1990 Tentang Kepariwisataaan. Tersediadi meta-puspita.blogspot.com/2012/03/undang-undang-republik-indonesia-no9\_18.html, Diakses 19 Agustus 2019.
- Putu, Wira Buana. 2010. *Penemuan Rute Terpendek pada Aplikasi Berbasis Web*, Lontar Komputer, Vol. 1, No. 1.
- R, Kumar dan M, Kumar. 2010. *Exploring Genetic Algorithm for Shortest Path Optimization in Data Networks*. *Global Journal of Computer Science and Technology*. Vol 10, Diakses 25 Februari 2020.
- Ramadhan, Zuhri. 2016. *Perbandingan Algoritma Prim dan Algoritma Floyd-Warshall dalam Menentukan Lintasan Terpendek (Shortest Path Problem)*. Tesis tidak diterbitkan. Medan: Fakultas Ilmu Komputer dan Teknik Informatika-Universitas Sumatera Utara.
- Siang, Jong J. 2014. *Riset Operasi dalam Pendekatan Algoritmis*. Edisi 2. C.V Andi Offset, Yogyakarta.
- Suhamdani, H. 2010. Analisis Pengembangan Pariwisata Alam Lewaja di Kabupaten Enrekang. *Jurnal Ilmu Pemerintahan*, 3 (2). Diakses 26 Juni 2019.
- Syahputra, S. 2017. Penentuan Rute Terpendek Pendistribusian Naskah Ujian Nasional Menggunakan Algoritma Djikstra (Dinas Pendidikan dan Pengajaran Kota Binjai). *Jurnal Teknik Informatika Kaputama (JTIK)*, 1(1), Diakses 28 Juni 2019.