

Aplikasi Pohon Merentang Minimum untuk Rancangan Rute Bus Wisata Bandung

Seperayo - 13516068

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jl. Ganेशha 10 Bandung 40132, Indonesia

13516068@std.stei.itb.ac.id

Abstrak—Kegiatan berwisata di suatu daerah memiliki sukaduka tersendiri. Salah satu duka yang paling tampak adalah ketika perencanaan kegiatan wisata itu tidak memperhatikan jarak antar tempat wisata, sehingga menyebabkan waktu yang terbuang di jalan. Penulis membuat makalah ini sebagai salah satu gambaran awal rute yang dapat diambil oleh Bus Wisata. Algoritma Prim yang dapat menghasilkan pohon merentang minimum digunakan untuk merepresentasikan rute yang dapat diambil.

Kata Kunci—Algoritma Prim, Rute, Graf.

I. PENDAHULUAN

Indonesia adalah salah satu negara yang terkenal akan sumber daya alamnya yang indah dan mempesona. Salah satu tempat yang paling banyak dikunjungi oleh wisatawan luar negeri adalah Kota Bali yang terkenal akan pantainya. Namun, Bali bukanlah satu-satunya kota yang layak untuk dikunjungi. Kota-kota lain di Indonesia juga memiliki daya tariknya sendiri, sayangnya masih banyak pengembangan yang perlu dilakukan demi terwujudnya Indonesia yang kaya akan pariwisata.

Kota Bandung atau yang juga dikenal sebagai Kota Kembang adalah salah satu tempat di Indonesia yang banyak dikunjungi oleh wisatawan, baik dari dalam maupun luar negeri. Secara domestik menurut data dari Tribun^[2], Bandung menjadi salah satu dari tiga kota yang paling banyak dikunjungi, kota lainnya adalah Yogyakarta dan Bali.

Dalam melakukan wisata banyak metode yang dilakukan oleh para wisatawan, ada yang memilih untuk menggunakan kendaraan pribadi, menggunakan jasa rental, atau menggunakan jasa umum yang disediakan oleh pemerintah daerah. Masing-masing pilihan memiliki kelebihan dan kekurangan masing-masing, seperti penggunaan kendaraan pribadi dapat menghemat biaya kendaraan.

Sedangkan penggunaan jasa rental kendaraan dapat memberi keuntungan dalam hal proses transportasi yang lebih nyaman dengan hanya duduk dan menikmati pemandangan sekitar, namun kekurangan metode ini adalah biaya yang dikeluarkan akan lebih tinggi dari metode pertama. Metode ketiga adalah menggunakan layanan umum, keuntungannya adalah informasi tempat wisata yang lebih lengkap dan gambaran kota yang lebih rinci, selain itu biaya yang dikeluarkan juga terbilang lebih terjangkau. Kelemahannya adalah privasi saat berwisata

kurang terjaga, dimana kita menikmati layanan ini bersama wisatawan lainnya.

Ketertarikan penulis terhadap gambaran keindahan Kota Bandung melalui layanan seperti Bus Wisata yang menawarkan informasi lengkap sekaligus nuansa Bandung yang indah ini yang memunculkan ide untuk membuat rute wisata di Bandung. Namun, pembuatan rute ini memerlukan berbagai macam pertimbangan dimulai dari waktu, jarak, sumber daya manusia, dst. Dalam makalah kali ini penulis hanya akan membahas dari aspek jarak. Bus Wisata sesuai dengan namanya perlu memberikan kepuasan maksimal bagi penumpangnya dalam menikmati keindahan Kota Bandung. Oleh sebab itu, diperlukan suatu rute yang efisien, dimana dapat mengantarkan penumpangnya ke berbagai tempat wisata dengan jarak yang minim, namun tetap memberikan nuansa Bandung sepenuhnya.



Gambar 1 — Bus Wisata di Bandung^[3]

II. TEORI DASAR

Dalam melakukan pemecahan masalah ini diperlukan beberapa pengetahuan dasar, yaitu :

2.1. Graf

Definisi umum dari graf adalah suatu himpunan yang terdiri dari sisi dan simpul $G = (V, E)$, dengan V adalah *vertices* dan E adalah *edge*.

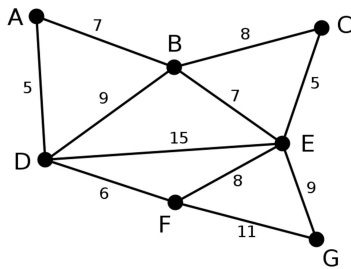
2.1.1. Jenis Graf

Graf sendiri dapat dibagi menjadi dua menurut arahnya, yaitu Graf Berarah (ditandai dengan adanya tanda panah di sisi graf) dan Graf Tak Berarah (graf yang tidak memiliki orientasi arah). Dalam makalah ini graf yang akan digunakan adalah Graf Tak Berarah.

2.1.2. Terminologi Graf

Terdapat beberapa terminologi yang perlu diketahui mengenai graf, seperti *Adjacent*

(ketetanggaan), *Degree* (derajat), *Path* (lintasan), *Circuit* (sirkuit), *Spanning Subgraph* (upagraf merentang), dan *Weighted Graph* (graf berbobot). Graf berbobot artinya graf yang memiliki nilai pada setiap sisi yang dimiliki, sedangkan upagraf merentang adalah suatu graf bagian atau upagraf yang memiliki seluruh simpul graf (terdapat sisi yang hilang). Sirkuit adalah lintasan yang memiliki titik awal dan titik akhir pada simpul yang sama, sedangkan lintasan sendiri berarti jalur yang dilalui di dalam graf dengan melalui beberapa simpul yang berselang-seling. Derajat adalah banyaknya sisi yang ada di suatu simpul, *Adjacent* artinya dua simpul itu saling terhubung oleh satu sisi yang sama secara langsung.



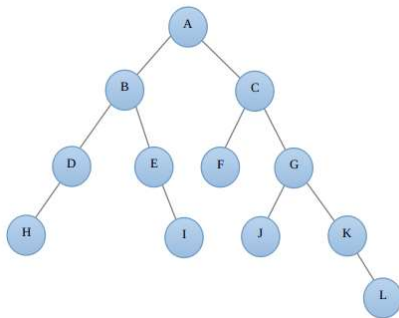
Gambar 2 — Graf Berbobot^[4]

2.2. Pohon

Pohon secara definisi merupakan suatu graf yang tak berarah, terhubung, dan tidak memiliki sirkuit.

2.2.1. Terminologi Pohon

Layaknya graf terminologi pohon juga hampir sama dengan terminologi graf. Perbedaannya adalah pohon memiliki daun, simpul dalam, tingkat, tinggi, dan hubungan anak-orangtua.



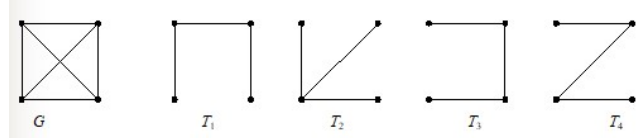
Gambar 3 — Pohon^[5]

Pada gambar diatas dapat dikatakan B dan C adalah anak A, A orangtua B dan C. F, H, I, J, K adalah daun (tidak memiliki anak), D, E, G, K, B, C adalah simpul dalam (memiliki anak), tingkat pohon diatas dimulai dari paling bawah adalah 4, 3, 2, 1, dan 0, sedangkan tinggi (tingkat maksimum) adalah 4.

2.2.2. Pohon Merentang (*spanning tree*)

Sama halnya dengan graf merentang, pohon merentang adalah pohon bagian dari pohon utama

yang memutus sirkuit di graf atau bisa disebut sebagai upagraf dari pohon utama.



Gambar 4 — Pohon Merentang^[1]

2.3. Algoritma Prim

Dalam membentuk suatu pohon merentang dikenal juga istilah pohon merentang minimum. Salah satu metode sederhana yang banyak digunakan adalah Algoritma Prim.

Algorithm 1: Standard Prim's algorithm.

Input: A non-empty connected weighted graph G composed of vertices V_G and edges E_G , possibly with null weights;
Result: The minimal spanning tree in the *finalPath* array;
Initialization: $V_T = \{r\}$, where r is a random starting node from V ;

```

while  $V_T \neq V_G$  do
  minimum  $\leftarrow \infty$ ;
  for Visited nodes  $s \in V_T$  do
    for all edges  $E(s,v)$  and  $v \notin V_T$  do
      if  $Weight(E) \leq minimum$  then
        minimum  $\leftarrow Weight(E)$ ;
        edge  $\leftarrow E$ ;
        newVisited  $\leftarrow v$ ;
  finalPath  $\leftarrow finalPath \cup \{edge\}$ ;
   $V_T \leftarrow V_T \cup \{newVisited\}$ ;

```

Gambar 5 — Pseudocode Algoritma Prim^[6]

Pseudocode diatas memiliki waktu asimptotik sebesar $O(n^2)$, berdasarkan struktur data yang diolah dapat pula divariasikan dengan menggunakan list *adjecency* dan fibonacci. Variasi ini akan menurunkan waktu asimptotik menjadi $O((V+E) \log V) = O(E \log V)$ dan $O(E + V \log V)$.^[6]

Algorithm 2: Second implementation of Prim's algorithm.

Input: A non-empty connected weighted graph G composed of vertices V_G and edges E_G , possibly with null weights;
Result: The minimal spanning tree in the *d* array;
Initialization: $V_T = \{r\}$ and $d[r] = 0$, where r is a random node from V ;

```

for  $v \in (V - V_T)$  do
  if  $E(r,v)$  then
     $d[v] \leftarrow Weight(E)$ ;
  else
     $d[v] \leftarrow \infty$ ;
while  $V_T \neq V_G$  do
  Find a vertex  $u$  such that:
  |  $d[u] = \min\{d(v) \mid v \in (V - V_T)\}$ 
   $V_T \leftarrow V_T \cup \{u\}$ ;
  for all  $v \in (V - V_T)$  do
     $d[v] \leftarrow \min\{d[v], Weight(u,v)\}$ ;

```

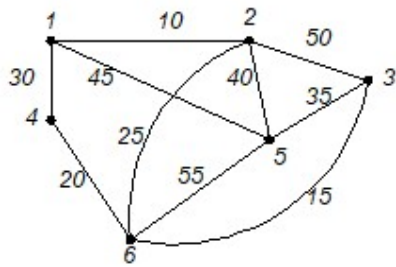
Gambar 6 — Pseudocode Algoritma Prim (Versi 2)^[6]

Secara sederhana pembentukan dari Algoritma Prim adalah sebagai berikut :

2.3.1. Ambil sisi dengan bobot terkecil

2.3.2. Pilih sisi dengan bobot paling minimum diantara 2 simpul dari sisi yang terakhir dipilih tanpa membentuk sirkuit.

2.3.3. Ulangi sebanyak (simpul-2) kali



Gambar 7 — Tabel contoh Prim^[1]

Gambar graf diatas kemudian dapat diolah sebagai berikut :

Step	Sisi	Bobot	Gambar
1	(1,2)	10	
2	(2,6)	25	
3	(3,6)	15	
4	(4,6)	20	
5	(3,5)	35	

Tabel 1 — Langkah-langkah dalam Algoritma Prim^[1]

III. TEMPAT WISATA KOTA BANDUNG

Terdapat beberapa data yang diperlukan dalam melakukan

pemecahan masalah dalam makalah ini, seperti

3.1. Lokasi Tempat Wisata di Bandung



Gambar 8 — Peta Bandung dan Tempat Wisatanya

Sumber : maps.google.com, diakses 03 Desember 2017 pukul 17.21 WIB.

3.2. Daftar Tempat Wisata

Label	Tempat
A	Farm House Susu Lembang
B	Floating Market Lembang
C	De' Ranch
D	Puncak Bintang Bandung
E	Bosscha Observatory
F	Tebing Keraton
G	Kawasan Punclut
H	Bird & Bromelia Pavilion
I	Andromeda High Park
J	Amazing Art World

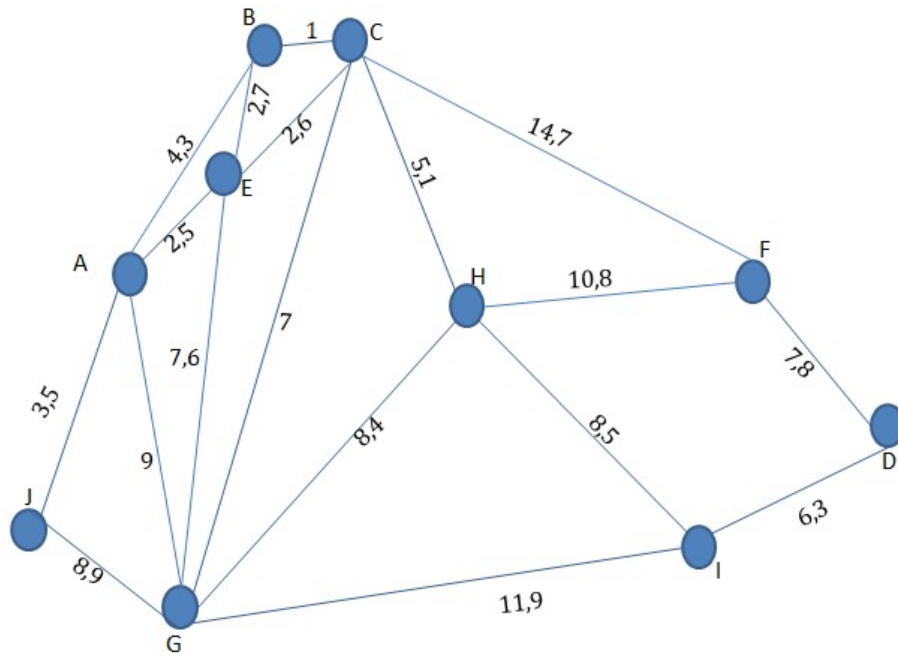
Tabel 2 — 10 Tempat Wisata Terbaik

Sumber : maps.google.com, diakses 03 Desember 2017 pukul 17.21 WIB.

Saya menggunakan data dari google yang mengurutkan tempat wisata berdasarkan *rating* pengunjung menjadi 10 tempat wisata terbaik. Alasan dipilihnya 10 tempat wisata dikarenakan keterbatasan waktu dan biaya dalam menentukan proses transportasi, dengan 10 tempat yang dipilih beban transportasi baik biaya ataupun lamanya liburan akan lebih terjangkau.

IV. PENENTUAN RUTE WISATA

Dalam memecahkan masalah pembentukan rute ini kita memerlukan sketsa graf dengan lokasi dari tempat-tempat wisata yang telah ditentukan, selain itu juga diperlukan penandaan bobot dan label masing-masing tempat wisata.



Gambar 9 — Graf Berbobot dari **Gambar 8**
 Sumber : pustaka pribadi

Graf pada gambar diatas dibuat dengan menggunakan pemahaman graf sebagai himpunan simbol dan sisi, bobot yang digambarkan dalam satuan kilometer dan merupakan jarak terdekat antar tempat yang mungkin dicapai tanpa merusak bentuk graf.

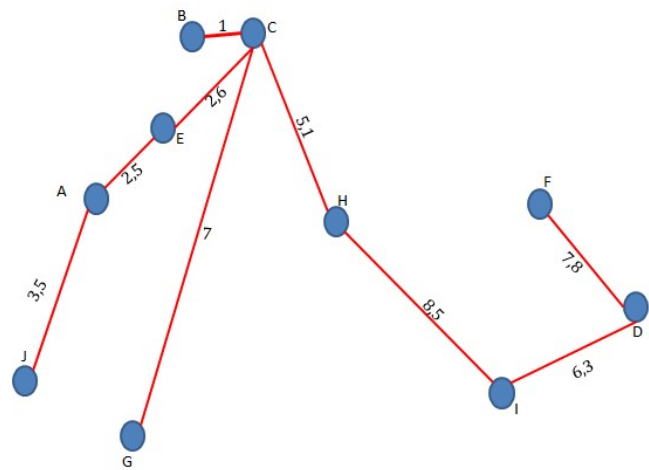
Step	Sisi	Bobot	Peta
1	(B,C)	1	
2	(C,E)	2,6	

3	(A,E)	2,5	
4	(A,J)	3,5	
5	(C,H)	5,1	

6	(C,G)	7	
7	(H,I)	8,5	
8	(D,I)	6,3	
9	(D,F)	7,8	
Total		44,3	

Tabel 3 — Operasi Pembentukan Rute Bus Wisata
Sumber : Pustaka Pribadi

Penggunaan Algoritma Prim dalam memecahkan masalah ini menemukan hasil total jarak tempuh minimal sebesar 44,3 Km. Terlihat bahwa dari jarak antar tempat wisata yang memiliki besaran antara 1 hingga 14,7 Km dengan jumlah tempat yang disinggahi sebanyak 10 buah, Bus Wisata hanya akan melalui jalan sepanjang 44,3 Km. Hal ini tentu akan menghemat waktu dan biaya transportasi saat wisata dijalankan.



Gambar 10 — Hasil Algoritma Prim dan Rute Paling Efisien
Sumber : pustaka pribadi

V. KESIMPULAN

Graf dan Pohon sebagai suatu teori yang dipelajari di Matematika Diskrit memiliki aplikasi yang luas, salah satunya adalah penentuan rute di peta. Rute hasil dari Algoritma Prim ini menghasilkan jarak efisien 44,3 Km untuk mengelilingi 10 tempat wisata di Kota Bandung.

Terdapat kelemahan pada penentuan rute dengan menggunakan cara ini, yaitu realisasinya yang masing minim. Hal ini disebabkan oleh masih banyaknya faktor lain yang perlu dipertimbangkan dalam menentukan rute paling efisien, seperti waktu buka masing-masing tempat wisata.

Metode ini cukup sulit digunakan jika data atau simbol yang ingin dimasukkan berjumlah banyak dan memiliki nilai bobot yang kompleks. Hal ini dikarenakan besarnya waktu asimptotik Algoritma Prim adalah $O(n^2)$.

VI. UCAPAN TERIMA KASIH

Pertama-tama penulis ingin mengucapkan puji dan syukur kepada Tuhan Yang Maha Esa atas anugerah dan bimbingan-Nya penulis dapat menyelesaikan makalah berjudul “Aplikasi Pohon Merentang Minimum untuk Rancangan Rute Bus Wisata Bandung” ini dengan baik dan lancar. Selain itu, penulis juga ingin mengucapkan terima kasih kepada Dosen pengajar Dra. Harlili M.Sc. dan Dosen pembimbing Dr. Ir. Rinaldi Munir, M.T., atas petunjuk dan arahan yang telah diberikan selama proses pengerjaan makalah ini berlangsung. Penulis juga tak lupa untuk berterima kasih kepada teman-teman yang memberikan masukan dan kritik selama penulis menyelesaikan makalah ini.

REFERENSI

- [1] R. Munir, “Pohon”, dalam *Matematika Diskrit*, ed. 3. Bandung:Informatika, 2010, hlm. 443-475.
- [2] Anon., *Inilah Tiga Kota di Indonesia Paling Banyak Dikunjungi Wisatawan Tahun Baru Ini*. 2015. [Online] Tersedia dalam: <http://jateng.tribunnews.com/2015/12/30/inilah-tiga-kota-paling-banyak-dikunjungi-wisatawan-tahun-baru-ini>. [diakses 2 Desember 2017 pukul 20.12 WIB].

- [3] Anon., *RAGAM BUS WISATA KOTA: JAKARTA PUNYA JCTB, BANDUNG PUNYA BANDROS*. 2014. [Online] Tersedia dalam: <https://www.hitsss.com/ragam-bus-wisata-kota-jakarta-punya-jctb-bandung-punya-bandros/>. [diakses 2 Desember 2017 pukul 19.48 WIB].
- [4] Matrisno, *Algoritma Prim*. 2015. [Online] Tersedia dalam: <http://matrisnowei.blogspot.co.id/2015/01/algoritma-prim.html>. [diakses 3 Desember 2017 pukul 08.34 WIB].
- [5] Anon., *Penelusuran Pohon Biner Algoritma DFS(Stack) dan Algoritma BFS(Queue)*. 2014. [Online] Tersedia dalam: <https://saungkode.files.wordpress.com/2014/04/pohonbiner.png>. [diakses 3 Desember 2017 pukul 09.04 WIB].
- [6] A. Mariano, *Hardware and Software Implementations of Prim's Algorithm for Efficient Minimum Spanning Tree Computation*. 2013. [Online] Tersedia dalam: https://link.springer.com/chapter/10.1007/978-3-642-38853-8_14. [diakses 3 Desember 2017 pukul 09.15 WIB].

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 3 Desember 2017



Seperayo - 13516068