Perencanaan Peta Tur Konser dengan Menggunakan Sirkuit Hamilton

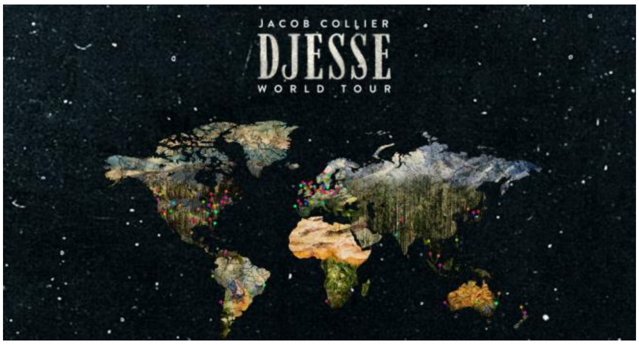
Muhammad Dehan Al Kautsar - 13519200[1]
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
[1]*13519200@std.stei.itb.ac.id*

***Abstract*—Dalam penentuan peta tur konser musisi, tentulah harus dipikirkan secara matang-matang dari segala aspek baik itu aspek jarak, biaya, serta keamanan dalam transportasi. Oleh karena itu dibutuhkan metode tertentu yang salah satunya adalah sirkuit Hamilton, yaitu sirkuit yang harus melewati semua simpul dalam graf sekali terkecuali simpul asal sebanyak dua kali. Dengan digunakannya penerapan sirkuit Hamilton pada masalah ini, kita dapat mencari rute yang terkecil dari segi biaya ataupun dari segi jarak antar kota meskipun penerapannya cukup sulit untuk dilakukan apabila kota atau simpul yang digunakan cukup banyak karena kompleksitas algoritma untuk mencari sirkuit Hamilton dengan beban terkecil.**

***Keywords*— Graf, Sirkuit Hamilton, Peta Tur Konser, Kompleksitas Algoritma**

I. PENDAHULUAN

Musisi dan tur konser merupakan dua hal yang sulit untuk dipisahkan. Pada umumnya, musisi mengeluarkan lagu baru mereka dengan skala waktu tertentu. Dalam kebutuhan promosi lagu, musisi tidak jarang melakukan sesuatu untuk menaikkan '*traffic'* mereka seperti kolaborasi dengan musisi lainnya untuk menarik hati penggemar dari musisi tersebut, merilis satu lagu kemudian satu album agar memberi kesan penasaran pada penggemarnya, atau mengadakan tur konser untuk meredam perasaan rindu penggemarnya.



Gambar 1. Tur Konser Salah Satu Musisi yang Gagal Dilakukan Karena Pandemi

(Sumber: https://www.jacobcollier.com/tour/ diakses pada 4 Desember 2020 pukul 18.44 WIB)

Dalam menyusun peta tur konser, sang perencana haruslah terampil dalam memilih rute antar kota agar semua kota yang direncanakan untuk diadakannya konser dapat dilalui dan kembali menuju kota asal musisi dengan jarak, waktu, ataupun biaya yang optimal. Hal ini bertujuan agar pendapatan yang didapatkan dapat maksimal dengan pengeluaran yang minimum. Namun untuk makalah ini akan dicari jarak paling minimum dalam perencanaan sebuah rute tur konser. Alasan lain dibutuhkannya rute dengan jarak atau waktu minimal adalah agar musisi tersebut tidak mengalami letih sebelum konser berlangsung. Sehingga sang musisi dapat kembali mempersiapkan waktu untuk menciptakan lagu lainnya yang menarik bagi penggemarnya.

II. LANDASAN TEORI

A. Graf

Graf adalah salah satu metode untuk merepresentasikan objek-objek diskrit dan hubungan antara objek-objek tersebut. Graf G memiliki dua buah komponen penting, yang biasanya dinotasikan sebagai G = (V,E) dengan V adalah *vortex* atau simpul dan E yang merepresentasikan *edge* atau sisi. Simpul dan sisi sendiri merupakan sebuah himpunan. Dalam membentuk graf, simpul tidak boleh merupakan himpunan kosong (setidaknya V = $\{V_1\}$). Graf sendiri biasanya dapat digambar dengan simpul merupakan titik atau lingkaran dengan garis yang menghubungkan antara titik tersebut yang disebut sisi.

Graf memiliki beberapa jenis, yaitu:

1. Graf sederhana
   Graf tidak berarah yang tidak memiliki sisi ganda dan sisi gelang,
2. Graf tidak sederhana
   Graf yang memiliki salah satu atau kedua-duanya dari sisi ganda maupun sisi gelang. Graf yang memiliki sisi ganda disebut graf ganda, sedangkan graf yang memiliki sisi ganda atau sisi gelang ataupun keduanya disebut graf semu.

Berdasarkan orientasi arahnya, graf juga dapat dibedakan menjadi 2 jenis, yaitu:

1. Graf tidak berarah
   Graf yang tidak memiliki orientasi arah, dan
2. Graf berarah
   Graf yang memiliki orientasi arah.

Kemudian, berikut adalah beberapa terminologi graf yang akan digunakan lebih lanjut dalam makalah ini:

1. Ketetanggan (*adjacent*
   Yaitu ketika kedua simpul terhubung langsung dengan salah satu sisi yang ada dalam graf.
2. Bersisian (*incidency*)
   Simpul V dikatakan bersisian dengan sisi E ketika sebuah simpul berhubungan langsung dengan sisi tersebut.
3. Simpul terpencil (*isolated vortex*)
   Simpul yang tidak memiliki sisi yang bersisian dengannya.
4. Graf kosong (*empty graph*)
   Suatu graf yang himpunan sisi nya merupakan himpunan kosong. (E = {})
5. Derajat (*degree*)
   Jumlah sisi yang bersisian dengan suatu simpul. Sebagai contoh simpul terpencil memiliki $d(V_1) = 0$, simpul yang memiliki satu sisi ganda memiliki $d(V_2) = 2$, dan simpul yang memiliki satu sisi gelang memiliki $d(V_3) = 2$.
6. Lintasan (*path*)
   Lintasan yang panjangnya n adalah jumlah sisi yang dilewati dari sebuah deretan simpul. Dalam lintasan juga terdapat istilah *minimum path* adalah jarak minimum yang harus dilalui suatu simpul untuk menuju simpul lainnya.
7. Sirkuit (*circuit*)
   Sirkuit adalah lintasan yang simpul asal dan simpul tujuannya sama. Sehingga lintasan berbentuk seperti tertutup, dan kembali ke simpul asal.
8. Keterhubungan
   Keterhubungan yaitu kondisi ketika dua simpul terdapat lintasan sehingga menghubungkan keduanya. Sedangkan graf disebut graf terhubung apabila untuk setiap pasang simpul v_1 dan v_2 dalam himpunan V terdapat lintasan dari v_1 menuju v_2. Jika tidak, maka disebut graf tak-terhubung.
9. Upagraf (*subgraph*)
   Upagraf yaitu suatu graf $G_1 = (V_1,E_1)$ yang jika $V_1 \subseteq V$ dan $E_1 \subseteq E$ dengan G = (V,E). Sementara itu komplemen graf adalah upagraf yang tidak menjadi bagian dari upagraf dari sebuah graf tersebut.
10. Upagraf merentang
    Upagraf merentang yaitu upagraf yang himpunan simpul nya masih utuh. Yaitu $V_1 = V$.
11. Cut-Set
    Cut-set yaitu himpunan dua simpul yang sisi-sisi di dalam himpunan tersebut dibuang dapat membuat graf menjadi dua komponen. Himpunan yang himpunan bagiannya merupakan cut-set bukanlah sebuah cut-set.
12. Graf berbobot.
    Graf bobot adalah graf yang tiap sisinya diberi harga (bobot).

Selain itu, terdapat beberapa graf khusus yang nantinya mungkin akan digunakan dalam pembuatan makalah ini, yaitu:

a. Graf lengkap, yaitu graf sederhana yang tiap simpulnya memiliki ketetanggaan dengan simpul lainnya, sehingga derajat tiap simpul adalah n-1 dengan n adalah banyaknya simpul.
b. Graf lingkaran, yaitu graf yang tiap simpulnya hanya memiliki 2 tetangga sehingga derajat tiap simpul adalah 2.
c. Graf teratur (*regular graph*) adalah graf yang tiap simpulnya memiliki derajat yang sama.

B. Representasi Graf

Terdapat berbagai cara untuk merepresentasikan graf, berikut adalah beberapa cara representasi graf:

1. Matriks Ketetanggaan (*adjacency matrix*) (yang akan digunakan kemudian di dalam makalah ini),
2. Matriks Bersisian (*incidency matrix*), dan
3. Senarai ketetanggaan (*adjacency list*).

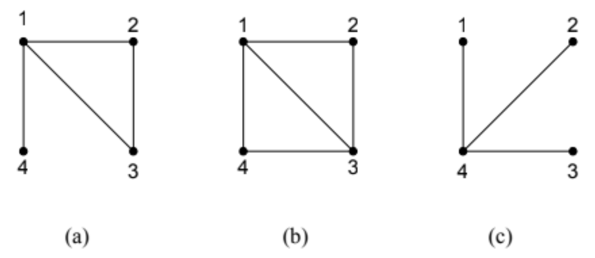
C. Lintasan dan Sirkuit Hamilton

Dalam aplikasinya, terdapat dua matematikawan yang berhasil mendefinisikan lintasan dan sirkuit di dalam graf. Ialah Leonhard Euler dan Sir William Rowan Hamilton yang berhasil memecahkan masalah diatas. Pada makalah ini hanya akan difokuskan kepada lintasan dan sirkuit Hamilton. Sedangkan lintasan dan sirkuit Euler tidak akan dibahas terlalu banyak pada makalah ini.

Lintasan Hamilton adalah lintasan yang melalui setiap simpul dalam sebuah graf. Tidak peduli apakah semua sisi sudah dilalui atau tidak dari lintasan tersebut, asalkan semua simpul telah dilalui maka lintasan tersebut dapat disebut lintasan Hamilton.

Sirkuit Hamilton sendiri adalah sirkuit yang melewati semua simpul di dalam graf sebanyak sekali, kecuali simpul asal yang akan dilalui sebanyak dua kali. Graf yang memiliki sirkuit Hamilton dinamakan graf Hamilton, sedangkan apabila hanya memiliki lintasan Hamilton dinamakan graf semi-Hamilton.

Syarat sebuah graf dapat dikatakan graf Hamilton sangat mudah untuk tercapai. Yaitu derajat tiap simpulnya haruslah n/2 (yaitu $d(v) \geq n/2$ untuk setiap simpul v di dalam graf G). Kemudian, menurut teorema, di dalam graf lengkap G dengan n buah simpul ($n \geq 3$), terdapat (n-1)!/2 kemungkinan sirkuit Hamilton.

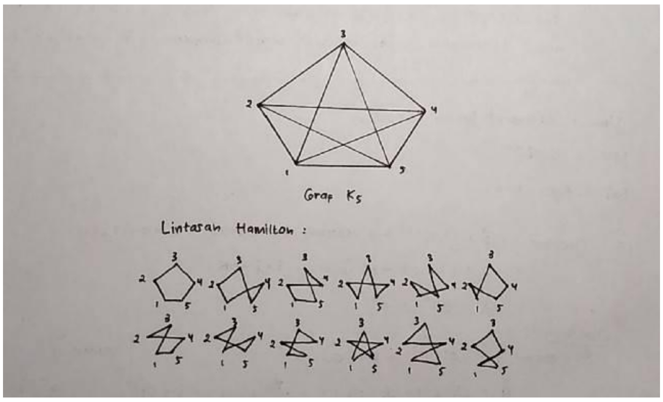


Gambar 2 Beberapa Contoh Graf

(Sumber: http://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian3.pdf diakses pada 4 Desember 2020 pukul 20.02 WIB)

Dari gambar diatas, dapat dianalisis satu-persatu untuk menentukan apakah graf tersebut dapat membentuk graf Hamilton atau tidak. Graf G_a tidak dapat membentuk graf Hamilton, namun terdapat lintasan Hamilton yaitu sebagai contoh lintasan 3,2,1,4. Graf G_b dapat membentuk graf Hamilton. Sebagai contoh sirkuit 2,3,4,1,2. Sedangkan graf G_c tidak dapat membuat lintasan ataupun sirkuit Hamilton.

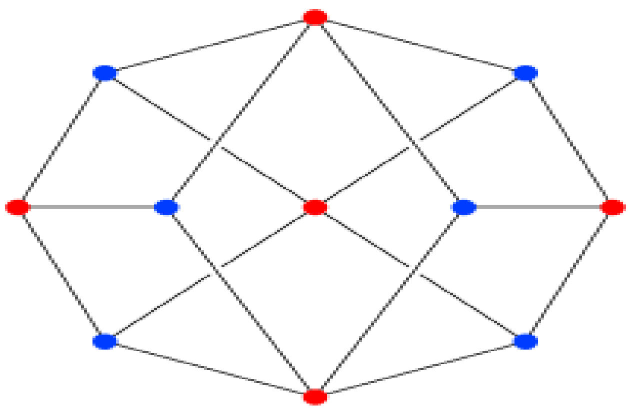
Berikut sebagai contoh merupakan sirkuit-sirkuit Hamilton untuk graf lengkap K_5.



Gambar 3 Sirkuit Hamilton Graf Lengkap K_5

(Sumber: Dokumen Pribadi)

Dalam membentuk graf, polihedral akan sangat mudah untuk mendapatkan sirkuit Hamilton. Hal itu dikarenakan banyaknya sisi yang dibutuhkan untuk membuat graf polihedral. Sampailah pada abad ke-18 seorang matematikawan Jerman Sir Friedrich William Harschel menemukan suatu polihedral terkecil yang dapat dibentuk agar tidak ditemukan adanya sirkuit Hamilton di dalam graf polihedral tersebut.



Gambar 4 Graf Herschel

(Sumber: https://aperiodical.com/2013/10/an-enneahedron-for-herschel/ diakses pada tanggal 5 Desember 2020 pukul 9.29)

III. ANALISIS KASUS

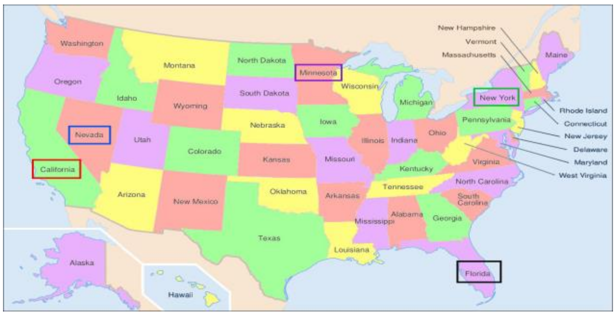
Diasumsikan terdapat seorang musisi, sebut saja Taylor yang berdiam di Los Angeles (kota di negara bagian California), ingin melakukan tur konser 5 kota di Amerika Serikat. Kota tersebut adalah Las Vegas (kota di negara bagian Nevada), Minnesota, New York, dan Miami (kota di negara bagian Florida). Taylor ingin mendapatkan pengeluaran transportasi terkecil ketika melakukan tur konser tersebut. Tugas seorang perencana tur untuk menentukan rute termurah berdasarkan biaya pesawat antar kota tersebut. (Asumsi Taylor menaiki pesawat komersil bertipe ekonomi dan harga tiket pesawat kota A ke kota B sama dengan tiket pesawat dari kota B ke kota A).

	LA	LV	Minnesota	NY	Miami
LA	-				
LV	28,94	-			
Minnesota	103,44	62,52	-		
NY	83,43	67,22	129,64	-	
Miami	82,32	57,43	156,30	49,95	-

Tabel 1 Harga Tiket Pesawat di Amerika Serikat (dalam US Dollar)

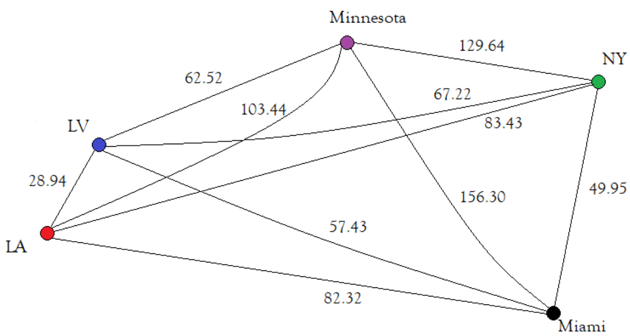
(Sumber: diolah dari https://google.com/flights/ diakses pada tanggal 4 Desember 2020 pukul 21.56)

Dapat dilihat dari tabel diatas, bahwa diantara kelima kota tersebut saling terkoneksi satu sama lain melalui transportasi udara. Sehingga dari tabel diatas dapat dibuat graf lengkap dengan jumlah simpul sebanyak lima.



Gambar 5 Kota-Kota yang Akan Dilalui Tur Konser

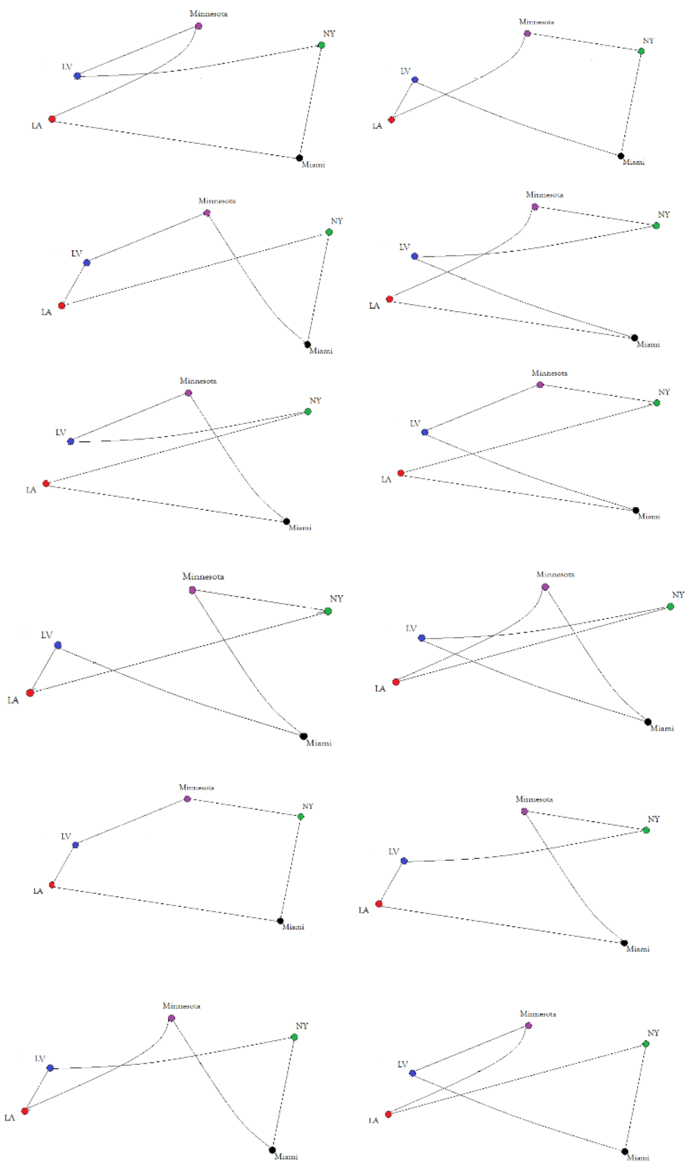
(Sumber: diolah dari https://elgl.org/diversity-dashboard-state-level-champions/ diakses pada tanggal 4 Desember 2020 pukul 22.08)



Gambar 6 Graf Antar 5 Kota Tur

(Sumber: Dokumen Pribadi)

Dengan diketahuinya graf berupa graf lengkap sebanyak 5 simpul, kita dapat membuat graf-graf Hamilton yang ada sebanyak (5-1)!/2 = 12 graf.



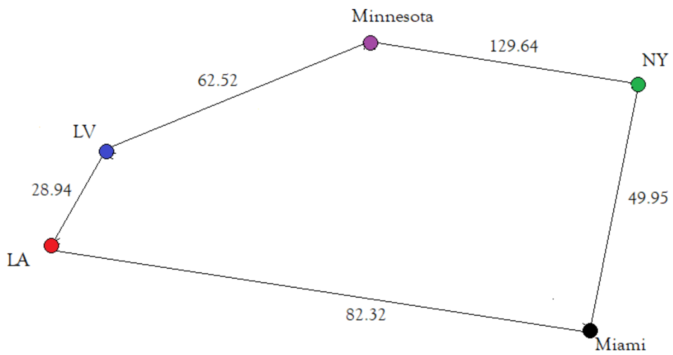
Gambar 7 Graf Merentang Hamilton yang dapat Dibentuk

(Sumber: Dokumen Pribadi)

Kumpulan gambar di atas adalah seluruh graf Hamilton yang dapat dibentuk dari peta tur konser pada Gambar 5. Untuk mencari jarak paling minimum kita harus menghitung beban dari tiap lintasan di atas. Berikut adalah perhitungan beban biaya per sirkuitnya:

1. Los Angeles – Las Vegas – Minnesota – New York – Miami – Los Angeles : 28,94 + 62,52 + 129,64 + 49,95 + 82,32 = 353,37
2. Los Angeles – Las Vegas – New York – Minnesota – Miami – Los Angeles : 28,94 + 67,22 + 129,64 + 156,30 + 82,32 = 464,42
3. Los Angeles – Las Vegas – New York – Miami – Minnesota – Los Angeles : 28,94 + 67,22 + 49,95 + 156,30 + 103,44 = 405,85
4. Los Angeles – Las Vegas – Miami – Minnesota – New York - Los Angeles : 28,94 + 57,43 + 156,30 + 129,64 + 83,43 = 455,74
5. Los Angeles – Las Vegas – Minnesota – Miami – New York - Los Angeles : 28,94 + 62,52 + 156,30 + 49,95 + 83,43 = 381,14
6. Los Angeles – Minnesota – Las Vegas – New York – Miami - Los Angeles : 103,44 + 62,52 + 67,22 + 49,95 + 82,32 = 365,45
7. Los Angeles – Las Vegas – Miami – New York – Minnesota - Los Angeles : 28,94 + 57,43 + 49,95 + 129,64 + 103,44 = 369,40
8. Los Angeles – Minnesota – New York – Las Vegas – Miami - Los Angeles : 103,44 + 129,64 + 67,22 + 57,43 + 82,32 = 440,05
9. Los Angeles – Minnesota – Las Vegas – Miami – New York - Los Angeles : 103,44 + 62,52 + 57,43 + 49,95 + 83,43 = 356,77
10. Los Angeles – New York – Las Vegas – Minnesota – Miami - Los Angeles : 83,43 + 67,22 + 62,52 + 156,30 + 82,32 = 451,79
11. Los Angeles – Minnesota – Miami – Las Vegas – New York - Los Angeles : 103,44 + 156,30 + 57,43 + 67,22 + 83,43 = 467,82
12. Los Angeles – New York – Minnesota – Las Vegas - Miami - Los Angeles : 83,43 + 129,64 + 62,52 + 57,43 + 82,32 = 415,34

Dari hasil perhitungan diatas didapatkan bahwa peta tur konser paling terbaik jatuh kepada rute Los Angeles – Las Vegas – Minnesota – New York – Miami – Los Angeles dengan biaya penerbangan total sebesar 353,37 Dolar Amerika Serikat.



Gambar 8 Rute Tur Konser dengan Pengeluaran Paling Sedikit (Los Angeles – Las Vegas – Minnesota – New York – Miami – Los Angeles)

(Sumber: Dokumen Pribadi)

IV. ANALISIS PROGRAM DAN KOMPLEKSITAS

```
tour-route.py ×
E: > IF > sem3 > Matematika Diskrit > makalah > tour-route.py > ...
# matriks ketetanggaan graf dibuat dalam graf di bawah
MatOfAdjGraph = [[0.0 for j in range (5)] for i in range (5)]

#graf haruslah graf lengkap
for i in range(5) :
    for j in range(5) :
        if (i<j):
            MatOfAdjGraph[i][j] = float(input("jarak dari kota "+str(i+1)+" ke kota "+str(j+1)+": "))
            MatOfAdjGraph[j][i] = MatOfAdjGraph[i][j] #kota 1 adalah kota asal

cond_two = False
cond_three = False
cond_four = False
cond_five = False
cond_six = False
cond_seven = False
cond_eight = False
cond_nine = False
cond_ten = False
cond_eleven = False
cond_twelve = False

temp = MatOfAdjGraph[0][1] + MatOfAdjGraph[1][2] + MatOfAdjGraph[2][3] + MatOfAdjGraph[3][4] + MatOfAdjGraph[4][0]
minimum_distance = temp
temp = MatOfAdjGraph[0][1] + MatOfAdjGraph[1][2] + MatOfAdjGraph[2][4] + MatOfAdjGraph[4][3] + MatOfAdjGraph[3][0]
if temp < minimum_distance:
    minimum_distance = temp
    cond_two = True
temp = MatOfAdjGraph[0][1] + MatOfAdjGraph[1][3] + MatOfAdjGraph[3][2] + MatOfAdjGraph[2][4] + MatOfAdjGraph[4][0]
if temp < minimum_distance:
    minimum_distance = temp
    cond_three = True
temp = MatOfAdjGraph[0][1] + MatOfAdjGraph[1][3] + MatOfAdjGraph[3][4] + MatOfAdjGraph[4][2] + MatOfAdjGraph[2][0]
if temp < minimum_distance:
    minimum_distance = temp
    cond_four = True
temp = MatOfAdjGraph[0][1] + MatOfAdjGraph[1][4] + MatOfAdjGraph[4][2] + MatOfAdjGraph[2][3] + MatOfAdjGraph[3][0]
if temp < minimum_distance:
    minimum_distance = temp
    cond_five = True
temp = MatOfAdjGraph[0][1] + MatOfAdjGraph[1][4] + MatOfAdjGraph[4][3] + MatOfAdjGraph[3][2] + MatOfAdjGraph[2][0]
```

```
tour-route.py ×
E: > IF > sem3 > Matematika Diskrit > makalah > tour-route.py > ...
temp = MatOfAdjGraph[0][1] + MatOfAdjGraph[1][4] + MatOfAdjGraph[4][3] + MatOfAdjGraph[3][2] + MatOfAdjGraph[2][0]
if temp < minimum_distance:
    minimum_distance = temp
    cond_six = True
temp = MatOfAdjGraph[0][2] + MatOfAdjGraph[2][1] + MatOfAdjGraph[1][3] + MatOfAdjGraph[3][4] + MatOfAdjGraph[4][0]
if temp < minimum_distance:
    minimum_distance = temp
    cond_seven = True
temp = MatOfAdjGraph[0][2] + MatOfAdjGraph[2][1] + MatOfAdjGraph[1][4] + MatOfAdjGraph[4][3] + MatOfAdjGraph[3][0]
if temp < minimum_distance:
    minimum_distance = temp
    cond_eight = True
temp = MatOfAdjGraph[0][2] + MatOfAdjGraph[2][3] + MatOfAdjGraph[3][1] + MatOfAdjGraph[1][4] + MatOfAdjGraph[4][0]
if temp < minimum_distance:
    minimum_distance = temp
    cond_nine = True
temp = MatOfAdjGraph[0][2] + MatOfAdjGraph[2][4] + MatOfAdjGraph[4][1] + MatOfAdjGraph[1][3] + MatOfAdjGraph[3][0]
if temp < minimum_distance:
    minimum_distance = temp
    cond_ten = True
temp = MatOfAdjGraph[0][3] + MatOfAdjGraph[3][1] + MatOfAdjGraph[1][2] + MatOfAdjGraph[2][4] + MatOfAdjGraph[4][0]
if temp < minimum_distance:
    minimum_distance = temp
    cond_eleven = True
temp = MatOfAdjGraph[0][3] + MatOfAdjGraph[3][2] + MatOfAdjGraph[2][1] + MatOfAdjGraph[1][4] + MatOfAdjGraph[4][0]
if temp < minimum_distance:
    minimum_distance = temp
    cond_twelve = True
```

```
tour-route.py ×
E: > IF > sem3 > Matematika Diskrit > makalah > tour-route.py > ...
temp = MatOfAdjGraph[0][3] + MatOfAdjGraph[3][2] + MatOfAdjGraph[2][1] + MatOfAdjGraph[1][4] + MatOfAdjGraph[4][0]
if temp < minimum_distance:
    minimum_distance = temp
    cond_twelve = True

print("Rute tur konser yang paling mangkus adalah", end=' ')
if cond_twelve:
    print("kota 1-4-3-2-5-1", end=' ')
elif cond_eleven:
    print("kota 1-4-2-3-5-1", end=' ')
elif cond_ten:
    print("kota 1-4-2-5-3-1", end=' ')
elif cond_nine:
    print("kota 1-3-4-2-5-1", end=' ')
elif cond_eight:
    print("kota 1-3-2-5-4-1", end=' ')
elif cond_seven:
    print("kota 1-3-2-4-5-1", end=' ')
elif cond_six:
    print("kota 1-2-5-4-3-1", end=' ')
elif cond_five:
    print("kota 1-2-5-3-4-1", end=' ')
elif cond_four:
    print("kota 1-2-4-5-3-1", end=' ')
elif cond_three:
    print("kota 1-2-4-3-5-1", end=' ')
elif cond_two:
    print("kota 1-2-3-5-4-1", end=' ')
else:
    print("kota 1-2-3-4-5-1", end=' ')
print("dengan jarak sebesar "+str(minimum_distance)+" km.")
```

Gambar 9 Program untuk Mencari Rute Tur Konser dengan Pengeluaran Terkecil

(Sumber : Dokumen Pribadi)

Program di atas adalah program yang dapat mencari tahu rute tur konser dengan pengeluaran terkecil terbatas untuk 5 kota. Untuk pencarian rute, penulis menggunakan rute-rute sirkuit Hamilton yang sudah dicari sebelumnya pada gambar 3 di atas. Untuk graf lengkap dengan jumlah simpul diatas 5, untuk mencari rute tur terpendek tampaknya cukup sulit dikarenakan sirkuit Hamilton yang dimungkinkan dibentuk dari graf lengkap K_6 saja adalah 60 sirkuit Hamilton. Untuk graf lengkap K_8 kita sudah harus membandingkan dari 2520 sirkuit Hamilton. Hal ini karena kompleksitas algoritma untuk mencari sirkuit Hamilton cukup buruk. Dapat dilihat waktu kompleksitas untuk membandingkan sirkuit Hamilton mana yang lebih terpendek dari n buah simpul saja sudah membutuhkan $T(n) = (n-1)!/2$. Hal ini menunjukkan bahwa kompleksitas algoritma ini untuk membandingkan lintasan Hamilton terpendek yaitu $\Theta(n!)$.

Sebenarnya masih banyak algoritma untuk menentukan sirkuit Hamilton terkecil ini. Seperti algoritma djikstra yang biasanya digunakan untuk mencari rute terpendek dari suatu tempat asal menuju destinasi yang dituju. Tapi memang algoritma *brute-force* lah yang sering digunakan untuk mencari rute-rute yang kemudian akan dibandingkan bebannya.

Berikut adalah percobaan yang dilakukan kepada program dengan kasus yang sama dengan analisis kasus di atas. Hasil menunjukkan hal serupa bahwa rute terpendek dari kasus di atas adalah graf lingkaran berurut 1-2-3-4-5-1 dengan panjang 353,37 km (Program ini membandingkan berdasarkan jarak, bukan berdasarkan biaya seperti program di atas).

```
PS C:\Users\asus> & C:/Users/asus/AppDa
jarak dari kota 1 ke kota 2: 28.94
jarak dari kota 1 ke kota 3: 103.44
jarak dari kota 1 ke kota 4: 83.43
jarak dari kota 1 ke kota 5: 82.32
jarak dari kota 2 ke kota 3: 62.52
jarak dari kota 2 ke kota 4: 67.22
jarak dari kota 2 ke kota 5: 57.43
jarak dari kota 3 ke kota 4: 129.64
jarak dari kota 3 ke kota 5: 156.30
jarak dari kota 4 ke kota 5: 49.95
Rute tur konser yang paling mangkus adalah kota 1-2-3-4-5-1 dengan jarak sebesar 353.37 km.
PS C:\Users\asus> 
```

Gambar 10 Hasil Analisis Kasus Menggunakan Program

(Sumber: Dokumen Pribadi)

Akan dicoba kembali menggunakan kasus lain. Digunakan tabel dibawah ini untuk mencari rute terpendek lainnya.

	London	Stoke	Cardiff	Manchester	Liverpool
London	-				
Stoke	157,4	-			
Cardiff	150,9	146,6	-		
Manchester	199,6	44,6	195,0	-	
Liverpool	211,7	57,0	207,4	34,3	-

Tabel 2 Jarak Antar 5 Kota di Inggris (dalam Mil)

(Sumber diolah dari https://www.google.com/ diakses pada tanggal 5 Desember 2020 pukul 10.44)

Gambar 11 Hasil Pencarian Rute Terpendek Berdasarkan Tabel 2

(Sumber: Dokumen Pribadi)

Didapatkan bahwa rute terpendek yang dapat dilakukan untuk mencapai sirkuit Hamilton adalah London – Cardiff – Stoke – Manchester – Liverpool – London dengan jarak 588,1 mil.

V. KESIMPULAN

Dari makalah ini, dapat disimpulkan bahwa rute tur konser seorang musisi akan mudah dicari dengan menggunakan konsep sirkuit Hamilton. Walaupun hal ini semakin sulit untuk rute dengan kota yang lebih dari lima dikarenakan kompleksitas algoritma yang semakin rumit. Namun apabila rute ini dapat ditemukan graf Hamilton minimumnya, kita dapat mendapatkan hasil yang optimal baik itu jarak maupun pengeluarannya.

Untuk makalah dengan tema serupa, penulis menyarankan untuk mengubah cara algoritma untuk mendapatkan sirkuit-sirkuit Hamilton karena cara yang penulis gunakan sangat boros waktu dan rentan dilakukukannya kesalahan.

VI. UCAPAN TERIMAKASIH

Penulis mengucapkan terima kasih kepada pihak-pihak yang telah mendukung penulis sehingga makalah yang dibuat dapat dikerjakan dengan lancar dan baik tanpa ada hambatan apapun. Terima kasih kepada Allah SWT yang telah memberi rahmat kepada penulis sehingga penulis dapat menyelesaikan makalah ini. Terima kasih pula untuk kedua orang tua penulis atas segala kasih sayang dan dukungan yang diberikan baik secara materil maupun non-materil.

Terima kasih kepada Bu Ulfa selaku dosen pengampu mata kuliah IF2120 Matematika Diskrit kelas K4 atas kerja kerasnya dalam mendidik penulis sehingga pengetahuan penulis cukup untuk membuat makalah ini. Terima kasih pula untuk Pak Rinaldi yang sudah membuat *website* yang sangat informatif sehingga pengerjaan makalah ini menjadi sangat terbantu. Tak lupa terima kasih untuk semua referensi yang penulis gunakan selama pembuatan makalah ini.

Terima kasih terakhir untuk teman-teman teknik informatika terutama teman-teman K4 dan beberapa teman di luar kelas K4 yang telah bersedia memberi saya saran-saran untuk pengerjaan makalah ini.

REFERENSI

[1] Munir, Rinaldi. 2016. *Matematika Diskrit*. Edisi Revisi Keenam. Bandung: Departemen Teknik Informatika.
[2] https://aperiodical.com/2013/10/an-enneahedron-for-herschel/ oleh Christian Lawson-Perfect, diakses 5 Desember 2020 pukul 9.25
[3] https://www.britannica.com/art/concert diakses pada 5 Desember 2020 pukul 11.26
[4] https://www.geeksforgeeks.org/mathematics-euler-hamiltonian-paths/ diakses pada 4 Desember 2020 pukul 23.37
[5] https://www.jacobcollier.com/tour diakses pada tanggal 4 Desember 2020 pukul 18.44

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.

Banyuwangi, 5 Desember 2020

Muhammad Dehan Al Kautsar
13519200