

Aplikasi Pewarnaan Graf untuk Menentukan Jadwal Maraton Menonton Film di Bioskop

Damianus Clairvoyance Diva Putra – 13520035¹

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

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

¹13520035@std.stei.itb.ac.id

Abstrak—Menonton film di bioskop merupakan sarana hiburan favorit banyak orang. Atas berbagai alasan, beberapa orang memilih untuk menonton film di bioskop secara maraton, yakni menonton beberapa film secara berturut-turut. Karena waktu tayang film di bioskop yang umumnya saling tumpang-tindih, akan digunakan konsep graf dan algoritma pewarnaan Welsh-Powell untuk menentukan kemungkinan jadwal yang tidak bertabrakan.

Kata kunci—maraton film bioskop, jadwal menonton film, graf, pewarnaan graf.

I. LATAR BELAKANG

Menonton film di bioskop masih menjadi pilihan hiburan paling populer masyarakat. Meskipun layanan aliran (*streaming*) film secara *on-demand* seperti Netflix makin marak, terutama di kala penutupan sementara bioskop selama pandemi Covid-19 belakangan ini, tak sedikit masyarakat masih menunggu kesempatan untuk menonton film di bioskop. Hal ini dibuktikan oleh laporan bertajuk “Social Distancing Moviegoing and TV Habits” oleh EDO tahun 2020, yang menemukan bahwa lebih dari 70 persen responden ingin segera menonton film layar lebar begitu bioskop beroperasi kembali.

Banyak orang pun tak jarang memutuskan untuk menonton film di bioskop secara maraton, yakni menonton beberapa film sekaligus dalam waktu yang berdekatan pada hari yang sama. Ada yang memilih untuk menonton film di bioskop secara maraton karena aksesnya terhadap bioskop terbatas, misalnya karena belum ada bioskop di sekitar tempat tinggal mereka. Menurut data Kemenparekraf, terhitung hingga Desember 2018, masih ada dua provinsi di Indonesia yang belum memiliki bioskop. Dari 32 provinsi lainnya, persebaran bioskop pun belum merata ke semua daerah sehingga banyak orang masih harus melakukan perjalanan ke bioskop terdekat di luar daerahnya. Ongkos transportasi dan konsumsi yang tinggi, serta kebutuhan waktu yang besar menjadi pertimbangan bagi mereka untuk menonton beberapa film sekaligus dalam satu kunjungan. Selain keterbatasan akses, ada pula yang memilih untuk menonton film di bioskop secara maraton karena memang merupakan seorang pencinta film.

II. DASAR TEORI

A. Teori Graf

Graf merupakan representasi objek-objek diskrit dan

hubungan antar-objek-objek diskrit tersebut. Secara matematis, graf didefinisikan sebagai $G = (V, E)$, dengan

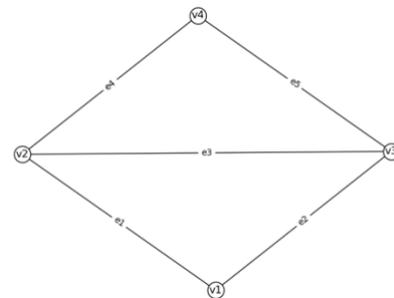
V = himpunan tidak kosong dari simpul (*vertices*)

$= \{v_1, v_2, \dots, v_n\}$

E = himpunan sisi (*edges*) yang menghubungkan dua simpul

$= \{e_1, e_2, \dots, e_n\}$

Dua sisi yang menghubungkan dua simpul yang sama disebut sisi ganda (*multiple edges* atau *parallel edges*), sedangkan sisi yang berawal dan berakhir pada simpul yang sama disebut sisi gelang atau kalang (*loop*). Sebuah graf digolongkan sebagai graf sederhana apabila graf tersebut tidak mengandung sisi ganda atau sisi gelang.



Gambar II.1. Graf Sederhana (Sumber: Arsip Penulis)

Dalam teori graf, terdapat beberapa terminologi yang akan memudahkan pembahasan selanjutnya, yakni sebagai berikut.

1. Bertetangga (*Adjacent*)

Dua simpul disebut bertetangga apabila keduanya terhubung langsung. Contohnya, untuk graf pada gambar II.1., simpul v_1 bertetangga dengan simpul v_2 melalui sisi e_1 dan dengan simpul v_3 melalui sisi e_2 , tetapi tidak bertetangga dengan simpul v_4 .

2. Bersisian (*Incidency*)

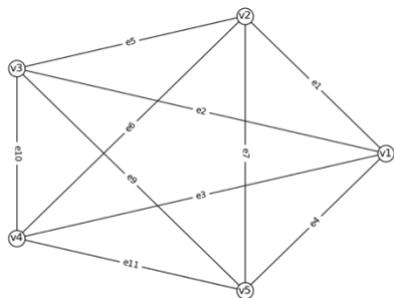
Untuk setiap $e = (v_i, v_j)$, sisi e disebut bersisian dengan simpul v_i dan bersisian dengan simpul v_j . Contohnya, untuk graf pada gambar II.1., sisi e_3 bersisian dengan simpul v_2 dan simpul v_3 .

3. Derajat (*Degree*)

Derajat suatu simpul didefinisikan sebagai jumlah sisi yang bersisian dengan simpul tersebut. Derajat simpul v dinotasikan sebagai $d(v)$. Contohnya, untuk graf pada gambar II.1., simpul v_1 dan v_4 memiliki derajat 2 ($d(v_1) = d(v_4) = 2$), sedangkan simpul v_2 dan v_3 memiliki derajat 3 ($d(v_2) = d(v_3) = 3$).

4. Graf Planar (*Planar Graph*)

Graf yang dapat digambarkan pada bidang datar dengan sisi-sisi tidak saling bersilangan disebut graf planar, sedangkan yang sebaliknya disebut graf tak-planar. Contohnya, graf pada gambar II.1. merupakan graf planar, sedangkan graf pada gambar II.2. merupakan graf tak-planar karena akan selalu terdapat dua sisi yang saling berpotongan.



Gambar II. 2. Graf Tidak Planar (Sumber: Arsip Penulis)

5. Graf Kosong (*Null Graph* atau *Empty Graph*)

Graf kosong didefinisikan sebagai graf yang himpunan sisinya merupakan himpunan kosong. Graf kosong dengan n simpul dinotasikan sebagai N_n .

6. Graf Lengkap (*Complete Graph*)

Graf lengkap didefinisikan sebagai graf sederhana yang setiap simpulnya memiliki sisi ke semua simpul lainnya. Graf lengkap dengan n simpul dinotasikan sebagai K_n .

7. Graf Lingkaran

Graf lingkaran didefinisikan sebagai graf sederhana yang setiap simpulnya berderajat dua. Graf lingkaran dengan n simpul dinotasikan sebagai C_n .

8. Graf Bipartite (*Bipartite Graph*)

Graf bipartite didefinisikan sebagai graf yang himpunan simpulnya dapat dipisah menjadi dua himpunan bagian V_1 dan V_2 , sedemikian sehingga setiap sisi pada graf menghubungkan sebuah simpul di V_1 ke sebuah simpul di V_2 . Graf bipartite dinotasikan sebagai $G(V_1, V_2)$.

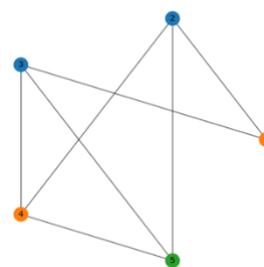
B. Pewarnaan Graf

1. Pewarnaan Simpul

Pewarnaan graf merupakan salah satu metode analisis graf untuk menyelesaikan berbagai persoalan, misalnya pewarnaan peta, penjadwalan, dan penyimpanan zat kimia. Pewarnaan graf dapat berupa pewarnaan simpul, sisi, atau wilayah. Untuk pewarnaan simpul, konsepnya ialah memberi warna setiap simpul pada graf sedemikian sehingga dua simpul yang bertetangga memiliki warna yang berbeda.

2. Bilangan Kromatik

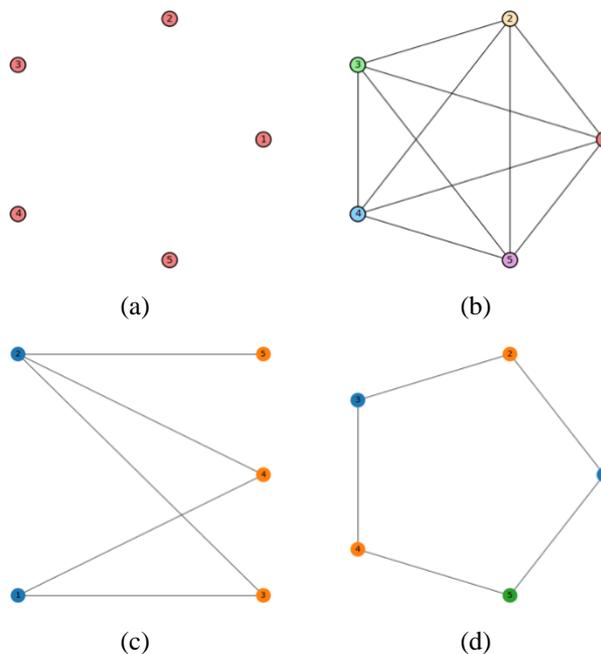
Bilangan kromatik didefinisikan sebagai jumlah minimum warna yang dibutuhkan untuk mewarnai peta. Bilangan kromatik dinotasikan sebagai $\chi(G) = k$ untuk graf G dengan bilangan kromatik k . Contohnya, graf G pada gambar II.3. memiliki $\chi(G) = 3$.



Gambar II.3. Pewarnaan Graf (Sumber: Arsip Penulis)

Dalam berbagai kasus khusus berikut, bilangan kromatis dapat langsung ditentukan dari jenis grafnya.

- (1) Graf kosong N_n memiliki $\chi(G) = 1$.
- (2) Graf lengkap K_n memiliki $\chi(G) = n$.
- (3) Graf bipartit $K_{m,n}$ memiliki $\chi(G) = 2$.
- (4) Graf lingkaran dengan n ganjil memiliki $\chi(G) = 3$.
- (5) Graf lingkaran dengan n genap memiliki $\chi(G) = 2$.
- (6) Pohon T memiliki $\chi(T) = 2$.
- (7) Graf planar memiliki $\chi(G) \leq 4$.



Gambar II.4. Pewarnaan Graf Kosong (a), Graf Lengkap (b), Graf Bipartite (c), dan Graf Lingkaran Simpul Ganjil (d) (Sumber: Arsip Penulis)

3. Algoritma Welsh-Powell

Algoritma Welsh-Powell merupakan salah satu algoritma pewarnaan graf dengan pendekatan *greedy*. Berikut merupakan langkah penerapan algoritma ini.

- (1) Hitung derajat dari setiap simpul dan urutkan nilainya dari terbesar sampai terkecil.
- (2) Beri warna untuk simpul paling atas urutan yang belum berwarna, lakukan iterasi terhadap simpul lainnya, lalu warnai pula simpul yang tidak bertetangga dengan simpul pertama dan belum berwarna.
- (3) Ulangi langkah (2) dengan warna yang berbeda sampai seluruh simpul telah diwarnai.

III. PEMBAHASAN

A. Analisis Contoh Kasus

Untuk keperluan analisis, akan digunakan jadwal tayangan film pada hari Minggu, 5 Desember 2021 di salah satu bioskop terbesar di Surabaya, yaitu Ciputra World XXI. Data untuk layar Regular 2D-nya disajikan dalam tabel berikut.

No.	Judul Film	Waktu Tayang (WIB)	Durasi (Menit)
1	Tadap	13:00	160
2		15:35	160
3		18:10	160
4		20:45	160
5	Ghostbusters: Afterlife	13:00	124
6		15:30	124
7		18:00	124
8		20:30	124
9	Raging Fire	12:40	126
10		15:15	126
11		17:50	126
12		20:25	126
13	Seperti Dendam, Rindu Harus Dibayar Tuntas	13:20	115
14		15:45	115
15		18:10	115
16	Eternals	20:35	115
17		13:15	156
18		16:35	156
19	Yowis Ben 3	19:40	156
20		13:05	113
21		15:30	113
22		17:55	113
23	Venom: Let There Be Carnage	20:20	113
24		13:25	97
25		15:35	97
26		17:45	97
27	The Boss Baby: Family Business	19:55	97
28		13:30	107
29		15:45	107
30		18:00	107
31		20:15	107

Tabel III.1. Data Jadwal Penayangan Film di Bioskop Ciputra World XXI Layar Regular 2D Tanggal 5 Desember 2021 (Sumber: <https://21cinplex.com/theater/>)

Untuk menyederhanakan lingkup pembahasan tanpa mereduksi kompleksitas persoalan, hanya akan dianalisis empat film pertama pada tabel tersebut, yakni data nomor 1 sampai 16. Perlu diingat bahwa tiap waktu tayang yang berbeda untuk judul film yang sama tetap dianggap sebagai komponen terpisah. Data pada tabel tersebut selanjutnya dapat direpresentasikan dalam bentuk graf dengan simpul menyatakan komponen dan sisi menyatakan hubungan dua komponen yang berkonflik, yaitu

- (1) dua simpul yang mewakili judul film yang sama, atau
- (2) dua simpul yang memiliki waktu tayang yang bertabrakan, baik sebagian maupun seluruhnya.

Dengan kata lain, dua simpul bertentangan jika keduanya mengalami konflik. Sebagai visualisasi konflik dua komponen, tiap komponen data dapat direpresentasikan dalam diagram blok dengan variabel waktu pada sumbu horizontal dan judul film pada sumbu vertikal.

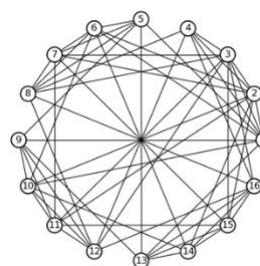


Gambar III.1. Visualisasi Konflik Antar-Komponen (Sumber: Arsip Penulis)

Sebagai contoh, komponen 1 mengalami konflik dengan komponen 2, 3, dan 4 karena memenuhi definisi konflik pertama (1), yaitu keempatnya merepresentasikan judul film yang sama. Selain itu, komponen 1 juga mengalami konflik dengan komponen 5, 6, 9, dan 13 karena memenuhi definisi konflik kedua (2), yaitu waktu tayangnya saling tumpang-tindih. Hasilnya dijabarkan dalam tabel berikut dan ditampilkan pada gambar III.2.. Graf termasuk graf sederhana dan graf tak-planar.

Simpul	Simpul Tetangga	Simpul	Simpul Tetangga
1	2, 3, 4, 5, 6, 9, 10, 13	9	1, 5, 10, 11, 12, 13
2	1, 3, 4, 6, 7, 10, 11, 14, 15	10	1, 2, 6, 9, 11, 12, 14
3	1, 2, 4, 7, 8, 11, 12, 15, 16	11	2, 3, 7, 9, 10, 12, 15
4	1, 2, 3, 8, 12, 16	12	3, 4, 8, 9, 10, 11, 16
5	1, 6, 7, 8, 9, 13	13	1, 5, 9, 14, 15, 16
6	1, 2, 5, 7, 8, 10, 14	14	2, 6, 10, 13, 15, 16
7	2, 3, 5, 6, 8, 11, 15	15	2, 3, 7, 11, 13, 14, 16
8	3, 4, 5, 6, 7, 12, 16	16	3, 4, 8, 12, 13, 14, 15

Tabel III.2. Simpul Graf dan Simpul Tetangganya.



Gambar III.2. Graf Representasi Persoalan (Sumber: Arsip Penulis)

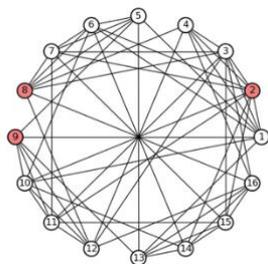
Untuk menentukan solusi jadwal film yang memungkinkan, akan diterapkan algoritma pewarnaan graf Welsh-Powell. Pertama, dihitung derajat tiap simpul dan diurutkan nilainya dari

terbesar hingga terkecil. Terdapat kemungkinan bahwa akan ada beberapa simpul yang memiliki derajat sama. Perbedaan urutan penulisan ini hanya akan berpengaruh pada hasil pewarnaan graf yang berbeda atau tidak unik.

Simpul	Derajat	Simpul	Derajat
2	9	12	7
3	9	15	7
1	8	16	7
6	7	4	6
7	7	5	6
8	7	9	6
10	7	13	6
11	7	14	6

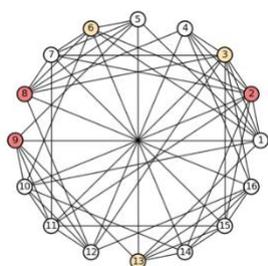
Tabel III.3. Derajat Setiap Simpul pada Graf.

Kedua, warnai simpul berderajat paling besar -dalam kasus ini simpul 2- dengan suatu warna, misalnya *lightcoral*, lalu lakukan iterasi terhadap simpul lainnya. Apabila simpul tersebut tidak bertetangga dengan simpul 2, warnai pula simpul tersebut dengan *lightcoral* -dalam kasus ini, simpul 8 dan 9.



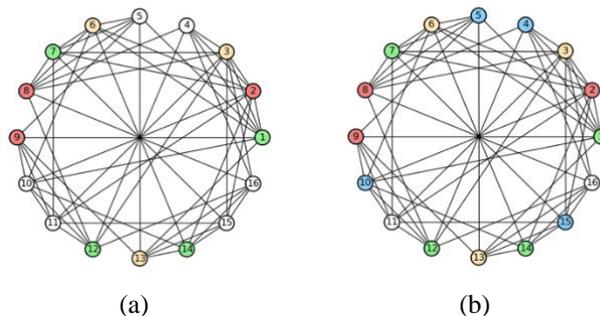
Gambar III.3. Graf Hasil Pewarnaan Pertama (Sumber: Arsip Penulis)

Selanjutnya, lakukan langkah yang sama untuk mewarnai sisa simpulnya. Sebagai contoh, untuk pewarnaan kedua, diketahui bahwa simpul 3 merupakan simpul berderajat paling tinggi yang belum berwarna. Maka, warnai simpul 3 dengan suatu warna lain, misalnya *moccasin*, kemudian warnai pula simpul 6 dan 13 dengan *moccasin* juga, karena simpul 3 tidak bertentangan dengan simpul 6 ataupun 13.



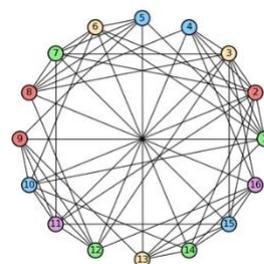
Gambar III.4. Graf Hasil Pewarnaan Kedua (Sumber: Arsip Penulis)

Pewarnaan ketiga mewarnai simpul 1, 7, 12, dan 14 dengan warna *lightgreen* dan pewarnaan keempat mewarnai simpul 10, 15, 4, dan 5 dengan warna *lightskyblue*.



Gambar III.5. Graf Hasil Pewarnaan Kedua (a) dan Pewarnaan Ketiga (b) (Sumber: Arsip Penulis)

Pewarnaan kelima mewarnai simpul 11 dan 16 dengan *plum*. Pada kondisi ini, semua simpul pada graf telah diproses dan algoritma Welsh-Powell dinyatakan selesai.



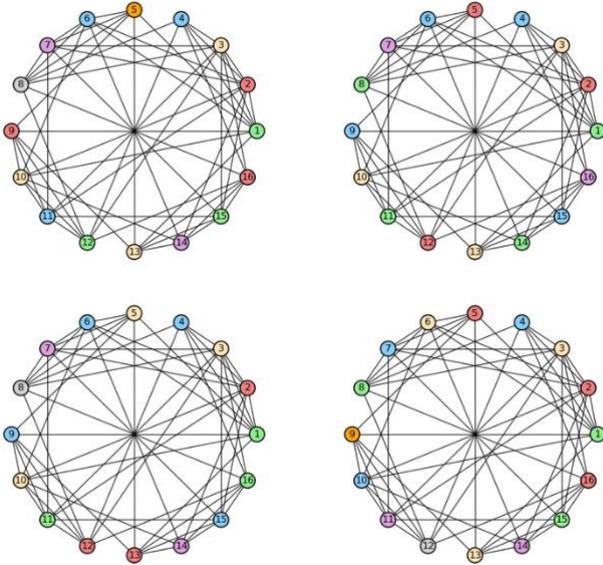
Gambar III.6. Graf Hasil Pewarnaan (Sumber: Arsip Penulis)

Dari graf tersebut, diperoleh bilangan kromatik sebesar 5 ($\chi(G) = 5$). Dengan kata lain, terdapat 5 kemungkinan jadwal menonton film yang dapat diturunkan dari hasil pewarnaan di atas. Untuk masing-masing warna yang sama, urutan simpul belum tentu mewakili urutan waktu sehingga jadwal masih harus diurutkan berdasarkan waktu.

Warna	Simpul	Waktu	Film
<i>Light-coral</i>	9	12:40-14:46	Raging Fire
	2	15:35-18:15	Tadap
	8	20:30-22:34	Ghostbusters: Afterlife
<i>Mocca-sin</i>	13	13:20-15:15	Seperti Dendam, Rindu Harus Dibayar Tuntas
	6	15:30-17:34	Ghosbusters: Afterlife
	3	18:10-20:50	Tadap
<i>Light-green</i>	1	13:00-15:40	Tadap
	14	15:45-17:40	Seperti Dendam, Rindu Harus Dibayar Tuntas
	7	18:00-20:04	Ghostbusters: Afterlife
<i>Light-skyblue</i>	12	20:25-22:31	Raging Fire
	5	13:00-15:04	Ghostbusters: Afterlife
	10	15:15-17:21	Raging Fire
<i>Plum</i>	15	18:10-20:05	Seperti Dendam, Rindu Harus Dibayar Tuntas
	4	20:45-23:25	Tadap
	11	17:50-19:56	Raging Fire
	16	20:35-22:30	Seperti Dendam, Rindu

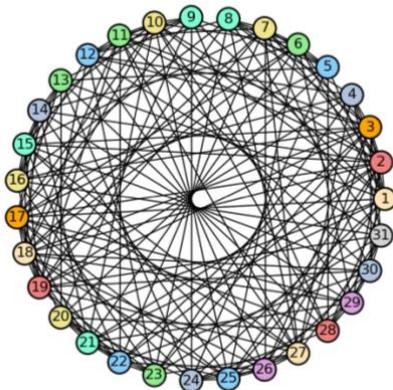
Tabel III.4. Lima Kemungkinan Jadwal Menonton Film.

Perhatikan bahwa seperti yang telah disebutkan di atas, hasil algoritma pewarnaan graf Welsh-Powell tidaklah unik karena terdapat beberapa simpul berderajat sama. Maka, urutan penulisan simpul yang berbeda pada tabel III.3 akan menghasilkan alternatif solusi lain, misalnya graf berikut.



Gambar III.7. Alternatif Lain Graf Hasil Pewarnaan (Sumber: Arsip Penulis)

Kembali lagi ke persoalan awal untuk menentukan jadwal dari 31 komponen, dengan dilakukannya proses yang sama untuk 16 komponen di atas, diperoleh salah satu alternatif hasil pewarnaan graf sebagai berikut.



Gambar III.8. Graf Hasil Pewarnaan untuk Persoalan Lengkap (Sumber: Arsip Penulis)

B. Eksplorasi Lanjutan dalam Program Python

Dari gambar III.6 dan III.8 di atas, dapat dilihat bahwa makin banyak pilihan film dan jam tayang, makin kompleks graf yang dimodelkan. Hal ini berimplikasi pada makin banyak pula langkah yang harus dilalui untuk memeriksa konflik antar-

simpul. Maka, pewarnaan graf secara manual tidak lagi efisien.

Oleh karena itu, dibuat program sederhana dengan bahasa Python untuk mengotomasi pencarian solusi dari masukan data apa pun, baik data besar maupun kecil. Dalam program ini, digunakan pustaka (*library*) Pandas, NetworkX, dan Matplotlib.

Pandas merupakan kakas (*tools*) sumber terbuka (*open source*) untuk menganalisis dan memanipulasi data dalam bahasa pemrograman Python. NetworkX merupakan *package* untuk membuat, memanipulasi, dan mempelajari struktur, dinamika, dan fungsi dari jaringan kompleks. Matplotlib merupakan pustaka untuk membuat visualisasi statis, teranimasi, dan interaktif dalam Python.

```
main.py
# Import library
import pandas as pd
import networkx as nx
import matplotlib.pyplot as plt
```

Dalam program bernama main.py ini, Pandas dimanfaatkan untuk membaca masukan data jadwal tayang film dari fail berformat .csv, misalnya dengan nama "movie_show.csv" [bagian (1)], yang berisi id, judul film, waktu tayang, dan durasi.

```
movie_show.csv
id,movie_name,start_time,duration
1,Tadap,1300,160
2,Tadap,1535,160
3,Tadap,1810,160
4,Tadap,2045,160,
5,Ghostbusters: Afterlife,1300,124
[redacted]
```

```
main.py
# Blok Pengolah Konflik
def conmin(time):
    # Mengonversi time (format HHMM) ke menit
    return time%100 + (time//100)*60

def get_element(data, id, element):
    # Mengambil nilai pada kolom element dan baris id
    value = data.loc[data["id"] == id][element].values
    return value

def is_conflict(data, id1, id2):
    # Menghasilkan True bila id1 dan id2 berkonflik

    # Konflik untuk id Beda
    start1 = conmin(get_element(data,id1,"start_time"))
    start2 = conmin(get_element(data,id2,"start_time"))
    duration1 = get_element(data,id1,"duration")
    duration2 = get_element(data,id2,"duration")
    end1 = start1 + duration1
    end2 = start2 + duration2
    conf_diff_id =
        (start2 <= start1 and end2 > start1) or
        (start2 >= start1 and start2 < end1)

    # Konflik untuk id Sama
    conf_same_id = get_element(data,id1,"movie_name")
    == get_element(data,id2,"movie_name")

    return conf_diff_id or conf_same_id

# Implementasi
movie_schedule = pd.read_csv("movie_show.csv") # (1)
```

NetworkX lalu diperlukan untuk memodelkan hasil olahan data tersebut dalam bentuk graf melalui prosedur Graph() [bagian (2)], add_nodes_from() [bagian (3)], dan add_edge() [bagian (4)]. Dalam prosedur add_edge(), terdapat proses pengolahan data untuk menemukan konflik antar-simpul.

```
main.py
# Blok Pengolah Graf
def add_nodes(data, network): # (3)
    # Menambahkan simpul sesuai data film
    list_id = data["id"].values
    network.add_nodes_from(list_id)
    return

def add_edges(data, network): # (4)
    # Menambahkan sisi antara dua simpul bila
    # kedua simpul tersebut mengalami konflik
    num = len(data)
    for i in range(1, num):
        for j in range(i+1, num+1):
            if (is_conflict(data, i, j)):
                network.add_edge(i, j)
    return

# Implementasi
show_network = nx.Graph() # (2)
add_nodes(movie_schedule, show_network) # (3)
add_edges(movie_schedule, show_network) # (4)
```

Kode algoritma Welsh-Powell tidak ditunjukkan di sini karena diadaptasi dari sumber referensi [13] [bagian (5)]. Data hasil olahan algoritma Welsh-Powell tersebut lalu diolah kembali oleh NetworkX, yaitu disusun posisinya [bagian (6)], diwarnai simpulnya [bagian (7)], dan digambarkan [bagian (8)]. Graf lalu diatur tampilannya oleh Matplotlib [bagian (9)].

```
main.py
# Implementasi
vtx_colors = graph_coloring(show_network, True) # (5)
pos = nx.circular_layout(show_network) # (6)
node_colors = [] # (7)
for n in show_network.nodes():
    node_colors.append(vtx_colors[n])

plt.rcParams["figure.figsize"] = [6, 6] # (9)

nx.draw_networkx( # (8)
    show_network, pos, nodelist = show_network.nodes(),
    edgelist = show_network.edges(),
    node_color = node_colors, node_size = 350,
    linewidths = 1.5, edgecolors = "black")

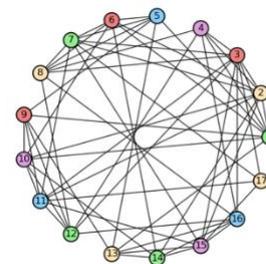
plt.axis('off') # (9)
plt.show()
```

C. Pengembangan Kasus

Metode ini dapat dikembangkan lebih lanjut sesuai dengan kebutuhan pengguna. Pertama, pengguna dapat memperoleh solusi jadwal untuk film yang memang ingin ditonton saja. Caranya ialah dengan memasukkan data film yang ingin ditonton dan menghapus data film lainnya dari fail .csv. Proses ini mirip seperti pengambilan empat film dari semua film ketika sedang menyederhanakan persoalan di atas.

Menonton film secara maraton memang terkadang melelahkan, maka sebagai pengembangan kedua, pengguna

dapat menambahkan slot waktu istirahat, yaitu rentang waktu ketika pengguna tidak ingin menonton film apa pun. Contohnya, apabila pengguna ingin istirahat makan pada pukul tujuh malam selama satu jam, dapat ditambahkan satu simpul baru pada graf -dalam kasus ini simpul 17- yang melambangkan slot istirahat tersebut. Graf barunya dimodelkan sebagai berikut.



Gambar III.9. Graf Hasil Pewarnaan Welsh-Powell dengan Slot Istirahat. (Sumber: Arsip Penulis)

Salah satu alternatif solusinya ialah mewarnai simpul istirahat dengan *moccasin*, bersamaan dengan simpul 2, 8, dan 13. Maka, salah satu jadwal yang bisa dipilih disajikan dalam tabel berikut.

Simpul	Waktu	Film
13	13:20-15:15	Seperti Dendam, Rindu Harus Dibayar Tuntas
2	15:35-18:15	Tadap
17	19:00-20.00	(Makan Malam)
8	20:30-22.34	Ghostbusters: Afterlife

Tabel III.5. Satu Kemungkinan Jadwal Menonton Film dengan Slot Istirahat.

V. KESIMPULAN

Konsep pewarnaan graf dapat digunakan untuk menyelesaikan persoalan penentuan jadwal menonton film di bioskop secara maraton. Dengan merepresentasikan tiap pilihan waktu tayang film sebagai simpul dan konflik antara dua pilihan sebagai sisi, lalu mewarnai setiap simpul sedemikian sehingga tidak ada simpul bertetangga yang berwarna sama, akan diperoleh jadwal yang tidak saling bentrok. Akan tetapi, besar kemungkinan akan terdapat banyak kemungkinan solusi sehingga pengguna tetap harus memilih salah satu yang paling menguntungkan baginya. Konsep ini juga dapat diterapkan untuk berbagai kasus lanjutan lainnya.

VII. PENUTUP

Penulis mengucapkan puji syukur kepada Tuhan Yang Maha Esa karena atas berkat dan rahmat-Nya, penulis dapat menyelesaikan makalah ini. Penulis juga tak lupa berterima kasih kepada semua pihak yang terlibat dalam penyusunan makalah ini, baik secara langsung maupun tidak. Penulis mengucapkan terima kasih kepada Bapak Rinaldi Munir selaku dosen pengampu mata kuliah IF2120 Matematika Diskrit Kelas 01 Semester 1 Tahun 2021/2022 yang telah membimbing penulis, serta kepada seluruh keluarga dan teman penulis yang telah mendukung penulis.

Penulis menyadari bahwa masih terdapat banyak kekurangan dalam penyusunan makalah ini. Oleh karena itu, penulis memohon maaf dan akan menerima secara terbuka setiap umpan balik yang membangun. Akhir kata, penulis berharap makalah ini dapat mendatangkan manfaat bagi banyak orang.

REFERENSI

- [1] Munir, R. (2021). Graf (Bag. 1) – Bahan Kuliah IF2120 Matematika Diskrit [slide powerpoint]. Diakses pada 13 Desember 2021, dari <https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf>
- [2] Munir, R. (2021). Graf (Bag. 2) – Bahan Kuliah IF2120 Matematika Diskrit [slide powerpoint]. Diakses pada 13 Desember 2021, dari <https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian2.pdf>
- [3] Munir, R. (2021). Graf (Bag. 2) – Bahan Kuliah IF2120 Matematika Diskrit [slide powerpoint]. Diakses pada 13 Desember 2021, dari <https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian3.pdf>
- [4] Bhattacharya, A. (2019). Welsh Powell Graph Colouring Algorithm. Diakses pada 13 Desember 2021, dari <https://www.geeksforgeeks.org/welsh-powell-graph-colouring-algorithm/>
- [5] 21Cineplex. (n.d.). Ciputra World XXI. Diakses pada 5 Desember 2021, dari <https://21cineplex.com/theater/bioskop-ciputra-world-xxi.313.SBYCIWO.htm>
- [6] Badan Ekonomi Kreatif (BEKRAF) dan filmindonesia.or.id (FI). (2019). Pemandangan Umum Industri Film Indonesia 2019. Diakses pada 13 Desember 2021, dari https://www.kememparekraf.go.id/asset_admin/assets/uploads/media/pdf/media_1589834208_PEMANDANGAN_UMUM_INDUSTRI_FILM_INDONESIA_2019.pdf
- [7] LIPI Press. (2021). Sirkulasi Film Platform Streaming di Indonesia. Diakses pada 13 Desember 2021, dari <https://lipipress.lipi.go.id/detailpost/sirkulasi-film-platform-streaming-di-indonesia>
- [8] Asrianti, S. Dan Rostanti Q. (2020). Studi: Penikmat Film Tak Sabar Kembali ke Bioskop. Diakses pada 13 Desember 2021, dari <https://nasional.republika.co.id/berita/q85c36425/studi-penikmat-film-tak-sabar-kembali-ke-bioskop>
- [9] Pandas. (n.d.). Pandas. Diakses pada 5 Desember 2021, dari <https://pandas.pydata.org/>
- [10] NetworkX. (n.d.). NetworkX: Network Analysis in Python. Diakses pada 5 Desember 2021, dari <https://networkx.org/>
- [11] Matplotlib. (n.d.). Matplotlib: Visualization with Python. Diakses pada 5 Desember 2021, dari <https://matplotlib.org/>
- [12] Boutellier, Y. (n.d.) Graph Coloring with NetworkX. Diakses pada 5 Desember 2021, dari <https://towardsdatascience.com/graph-coloring-with-networkx-88c45f09b8f4>
- [13] Tinoco, A.S. (2021). Graphs with NetworkX. Diakses pada 5 Desember 2021, dari <https://ansecura7.github.io/Algorithms/graphs/Graphs.html>

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.

Surabaya, 14 Desember 2021



Damianus Clairvoyance Diva Putra - 13520035