

Optimasi Penjadwalan Penyampaian Materi dalam Kaderisasi di Institut Teknologi Bandung dengan Algoritma Kahn

Ahmad Syafiq 13523135¹
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganessa 10 Bandung 40132, Indonesia
ahmad.syafiq2005@gmail.com, 13523135@std.stei.itb.ac.id¹

Abstrak—Penjadwalan penyampaian materi dalam kaderisasi di Institut Teknologi Bandung (ITB) sering menghadapi tantangan, salah satunya karena dependensi logis antarmateri. Dalam makalah ini, diusulkan algoritma Kahn, metode penyortiran topologis berbasis graf berarah, untuk mengoptimalkan jadwal penyampaian materi. Studi kasus dilakukan pada proses kaderisasi di Skhole-ITB Mengajar. Hasil implementasi menunjukkan algoritma mampu menghasilkan urutan logis yang efisien dan bebas kesalahan manual. Namun, algoritma ini kurang mempertimbangkan pengelompokan materi serupa. Solusi berupa pengelompokan materi berdasarkan kesamaan konteks diusulkan untuk menghasilkan jadwal yang lebih praktis.

Kata Kunci—Algoritma Kahn, Graf Asiklik Berarah, Kaderisasi, Penjadwalan.

I. PENDAHULUAN

Kaderisasi merupakan salah satu proses penting dalam pembentukan karakter, peningkatan kompetensi, dan transfer nilai-nilai organisasi kepada anggota baru. Di Institut Teknologi Bandung (ITB), kaderisasi menjadi bagian integral dari pengembangan mahasiswa, terutama dalam organisasi kemahasiswaan seperti Unit Kegiatan Mahasiswa (UKM) dan Badan Semi Otonom (BSO). Dalam proses kaderisasi, salah satu kegiatan utama yang sangat krusial adalah penyampaian materi kepada para kader. Penjadwalan penyampaian materi ini memegang peranan penting karena berfungsi sebagai sarana untuk menurunkan nilai-nilai organisasi dan membentuk profil kader yang sesuai dengan kebutuhan masing-masing organisasi.

Namun, penjadwalan penyampaian materi sering kali menghadapi tantangan. Proses kaderisasi dituntut untuk berhasil menurunkan semua materi untuk pemenuhan profil para kader sesuai kebutuhan masing-masing organisasi dalam waktu yang terbatas. Terlebih lagi, beberapa materi yang disampaikan memiliki keterkaitan logis dengan materi yang lain sehingga penyampaian materi tersebut tidak akan efektif tanpa didahului oleh penyampaian materi lain yang menjadi prasyaratnya.

Penjadwalan yang tidak optimal dapat menimbulkan sejumlah masalah, seperti ketidakteraturan dalam urutan

penyampaian materi, pengulangan materi yang tidak diperlukan, serta kurangnya pemahaman di kalangan peserta. Dalam banyak kasus, keterbatasan waktu dan sumber daya sering kali memaksa panitia kaderisasi untuk menyusun jadwal secara manual. Pendekatan manual ini tidak hanya memakan waktu yang signifikan tetapi juga rentan terhadap kesalahan, baik dalam penjadwalan urutan materi maupun dalam alokasi waktu yang efisien.

Oleh karena itu, diperlukan sebuah pendekatan yang lebih terstruktur dan efisien untuk menyusun jadwal penyampaian materi dalam kaderisasi. Dengan menggunakan metode yang terencana dan didukung oleh algoritma penjadwalan yang tepat, proses penyusunan jadwal dapat dioptimalkan sehingga tidak hanya mengurangi beban kerja panitia tetapi juga meningkatkan efektivitas penyampaian materi kepada para kader.

Algoritma Kahn, yang dikenal sebagai algoritma topological sorting, menawarkan solusi yang potensial untuk mengatasi masalah ini. Dengan menggunakan representasi graf terarah, algoritma ini mampu menentukan urutan optimal dari suatu rangkaian aktivitas berdasarkan keterkaitan atau ketergantungan antarelemen. Dalam konteks kaderisasi, algoritma Kahn dapat digunakan untuk memastikan bahwa materi-materi dengan ketergantungan logis disampaikan dalam urutan yang benar, sehingga peserta dapat memahami materi secara bertahap dan terstruktur.

Makalah ini bertujuan untuk mengembangkan model optimasi penjadwalan penyampaian materi dalam proses kaderisasi di ITB, khususnya pada salah satu BSO yang bergerak di bidang pengabdian masyarakat sektor pendidikan, yaitu Skhole-ITB Mengajar. Model ini diharapkan dapat membantu panitia kegiatan kaderisasi dalam menyusun jadwal yang efisien dengan memanfaatkan algoritma Kahn untuk mengoptimalkan urutan penyampaian materi.

Struktur makalah ini sebagai berikut: Bagian II akan membahas landasan teori terkait algoritma Kahn dan implementasinya dalam penjadwalan. Bagian III menjelaskan metodologi yang digunakan dalam penelitian ini. Bagian IV membahas hasil eksperimen dan analisisnya. Bagian V menyimpulkan temuan utama serta memberikan rekomendasi untuk penelitian lebih lanjut. Bagian VI berisi lampiran dan

Bagian VII berisi ucapan terima kasih.

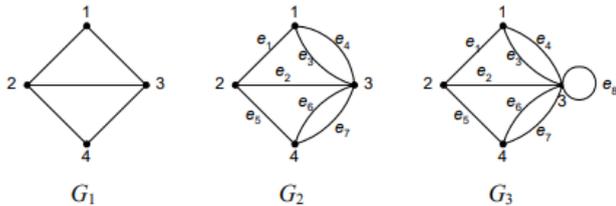
II. LANDASAN TEORI

A. Graf

Graf adalah hubungan antara objek-objek diskrit yang direpresentasikan dengan simpul dan sisi. Graf G didefinisikan sebagai $G = (V, E)$, yang dalam hal ini:

V = himpunan tidak-kosong dari simpul-simpul (*vertices*) = $\{v_1, v_2, \dots, v_n\}$

E = himpunan sisi (*edges*) yang menghubungkan sepasang simpul = $\{e_1, e_2, \dots, e_n\}$

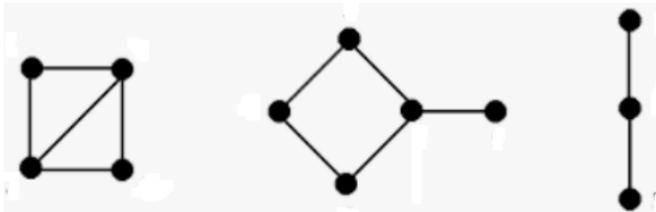


Gambar 2.1. (a) graf sederhana, (b) graf ganda, dan (c) graf semu (sumber: Website Rinaldi Munir) diakses pada 1 Januari 2025

Graf dapat digolongkan menjadi dua jenis berdasarkan ada atau tidaknya gelang atau sisi ganda pada suatu graf.

1. Graf sederhana (*simple graph*)

Graf yang tidak mengandung gelang maupun sisi ganda dinamakan graf sederhana.

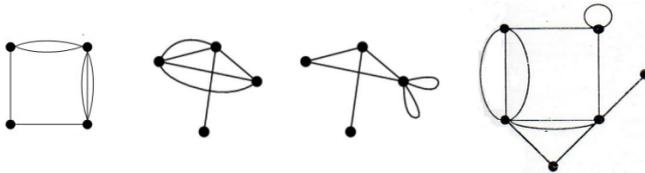


Gambar 2.2. Graf sederhana

(sumber: Website Rinaldi Munir) diakses pada 1 Januari 2025

2. Graf tak-sederhana (*unsimple-graph*)

Graf yang mengandung sisi ganda atau gelang dinamakan graf tak-sederhana (*unsimple graph*).



Gambar 2.3. Graf tak-sederhana

(sumber; Website Rinaldi Munir) diakses pada 1 Januari 2025

Graf tak-sederhana dibedakan lagi menjadi:

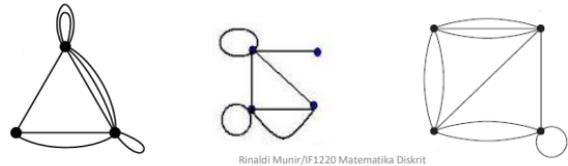
1. Graf ganda (*multi-graph*) yaitu graf yang mengandung sisi ganda



Gambar 2.4. Graf ganda

(sumber; Website Rinaldi Munir) diakses pada 1 Januari 2025

2. Graf semu (*pseudo-graph*) yaitu graf yang mengandung sisi gelang



Rinaldi Munir/IF1220 Matematika Diskrit

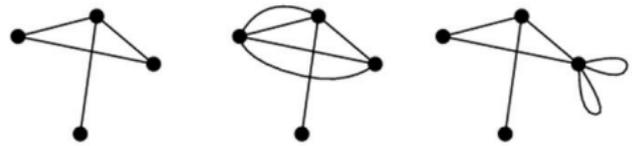
Gambar 2.5. Graf semu

(sumber: Website Rinaldi Munir) diakses pada 1 Januari 2025

Berdasarkan orientasi arah pada sisi, graf dibedakan atas dua jenis:

1. Graf tak-berarah (*undirected graph*)

Graf yang sisinya tidak mempunyai orientasi arah disebut graf tak-berarah.

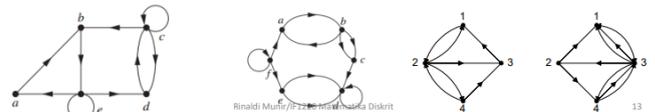


Gambar 2.6. Graf tak-berarah

(sumber: Website Rinaldi Munir) diakses pada 1 Januari 2025

2. Graf berarah (*directed graph* atau *digraph*)

Graf yang setiap sisinya diberikan orientasi arah disebut sebagai graf berarah.



Rinaldi Munir/IF1220 Matematika Diskrit

Gambar 2.7. Graf berarah

(sumber: Website Rinaldi Munir) diakses pada 1 Januari 2025

Graf berarah yang tidak memuat siklus berarah disebut graf asiklik berarah (DAG).

Graf berbobot adalah graf yang sisinya memiliki sebuah nilai yang mempunyai arti.

Graf lengkap adalah graf sederhana yang setiap simpulnya mempunyai sisi ke semua simpul lainnya.

Graf lingkaran adalah graf sederhana yang setiap simpulnya berderajat dua.

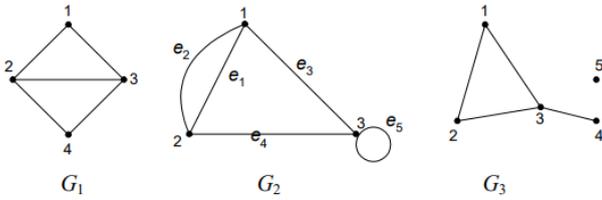
Graf teratur adalah graf yang setiap simpulnya mempunyai derajat yang sama

Graf bipartite adalah 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 .

Terminologi di dalam Graf:

1. Ketetanggaan (*Adjacent*)

Dua buah simpul dikatakan bertetangga bila keduanya terhubung langsung.



Gambar 2.8. Ketetangaan pada graf
(sumber: Website Rinaldi Munir) diakses pada 1 Januari 2025

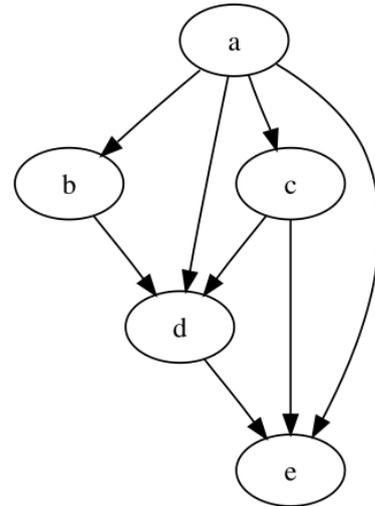
2. Bersisian (*Incidency*)
Suatu sisi akan bersisian dengan kedua simpul pada ujungnya.
3. Simpul terpencil (*Isolated Vertex*)
Simpul terpencil ialah simpul yang tidak mempunyai sisi yang bersisian dengannya.
4. Graf Kosong (*null graph*)
Graf yang himpunan sisinya merupakan himpunan kosong.



Gambar 2.9. Graf kosong
(sumber: Website Rinaldi Munir) diakses pada 1 Januari 2025

5. Derajat
Derajat suatu simpul adalah jumlah sisi yang bersisian dengan simpul tersebut.
Notasi: $d(v)$
6. Lintasan (*path*)
Sebuah urutan berurutan yang bergantian antara simpul dan sisi, yang disusun sedemikian rupa sehingga dimulai dan diakhiri oleh simpul, serta setiap sisi dalam urutan tersebut berhubungan dengan simpul sebelumnya dan sesudahnya.
7. Siklus (*Cycle*)
Lintasan yang berawal dan berakhir pada simpul yang sama disebut siklus
8. Keterhubungan
Sebuah graf dianggap terhubung jika setiap pasangan simpul yang dipilih secara acak dalam graf tersebut terhubung.
9. Upagraf
Upagraf adalah himpunan simpul dan sisi yang merupakan himpunan bagian dari sebuah graf. Suatu upagraf dikatakan sebagai upagraf merentang dari suatu graf jika upagraf itu mengandung semua simpul dari graf tersebut.

10. Cut-Set
Himpunan sisi yang bila dibuang dari G menyebabkan G tidak terhubung
11. Graf Berbobot
Graf berbobot adalah graf yang setiap sisinya diberi sebuah harga (bobot).
12. Graf Asiklik Berarah (*Directed Acyclic Graph*)
Graf Asiklik Berarah, atau DAG (*Directed Acyclic Graph*), adalah struktur dalam teori graf yang terdiri dari sekumpulan simpul (*vertices*) dan sisi (*edges*) berarah sedemikian rupa sehingga tidak ada lintasan tertutup berarah (*directed cycle*).



Gambar 2.9. Graf asiklik berarah
(sumber: Wikipedia) diakses pada 1 Januari 2025

Secara formal, DAG adalah graf berarah $G = (V, E)$, dengan V sebagai himpunan simpul dan E sebagai himpunan sisi, di mana setiap sisi $(u, v) \in E$ memiliki arah dari simpul u ke simpul v . DAG memiliki sifat berikut:

1. Tidak Memiliki Siklus Berarah: Tidak ada rangkaian sisi v_1, v_2, \dots, v_k ($k \geq 2$) di mana $v_i = v_k$, dan setiap sisi (v_i, v_{i+1}) memiliki arah $v_i \rightarrow v_{i+1}$.
2. Penataan Topologis: DAG selalu dapat diurutkan secara topologis, yaitu penyusunan simpul-simpulnya dalam urutan linear sedemikian rupa sehingga untuk setiap sisi (u, v) , simpul u muncul sebelum v dalam urutan tersebut.
3. Model Sebab-Akibat: DAG sering digunakan untuk merepresentasikan hubungan sebab-akibat atau dependensi, misalnya dalam grafik alir data, struktur hierarkis, dan model probabilistik.

B. Penyortiran Topologis

Penyortiran topologis adalah urutan linear dari simpul-simpul (*vertices*) dalam sebuah graf asiklik berarah, sedemikian rupa sehingga untuk setiap sisi berarah $u \rightarrow v$, simpul u muncul sebelum simpul v dalam urutan tersebut.

Penyortiran topologis berguna dalam berbagai aplikasi seperti pengelolaan proyek, kompilasi kode, dan penjadwalan tugas dengan dependensi.

Dalam masalah penjadwalan dengan dependensi, terdapat tugas-tugas yang harus dilakukan dalam urutan tertentu berdasarkan ketergantungan antara tugas.

Graf DAG digunakan untuk memodelkan masalah ini dengan menggunakan simpul yang merepresentasikan tugas-tugas serta sisi berarah menunjukkan dependensi antara tugas-tugas.

Misalnya, tugas A, B, C , dengan ketergantungan $A \rightarrow B \rightarrow C$. Dalam hal ini, tugas A harus selesai sebelum B dimulai, dan tugas B harus selesai sebelum C dimulai.

Algoritma Kahn adalah salah satu metode untuk melakukan penyortiran topologis pada DAG. Algoritma ini berbasis pada konsep derajat masuk, yaitu jumlah sisi yang masuk ke sebuah simpul. Algoritma Kahn dilakukan dengan:

1. Hitung derajat masuk: Tentukan derajat masuk untuk setiap simpul dalam graf.
2. Identifikasi simpul tanpa dependensi: Masukkan semua simpul dengan derajat masuk 0 ke dalam sebuah antrian (*queue*). Simpul-simpul ini dapat langsung diproses karena tidak memiliki tugas pendahulu.
3. Proses antrian: Ambil simpul dari antrian dan tambahkan ke dalam urutan topologis. Kemudian, untuk setiap simpul yang bertetangga (sisi keluar), kurangi derajat masuknya sebesar 1. Jika derajat masuk menjadi 0, masukkan simpul tersebut ke dalam antrian.
4. Ulangi proses hingga antrian kosong

III. METODOLOGI

Metodologi yang digunakan dalam makalah ini mencakup beberapa tahap utama, yaitu perancangan model, implementasi algoritma, dan evaluasi hasil. Adapun penjelasan tiap tahap adalah sebagai berikut:

A. Perancangan Model

Data yang digunakan diambil dari materi yang diajarkan dalam proses kaderisasi pada Skhole-ITB Mengajar. Selain itu, identifikasi dependensi antara materi dilakukan secara manual untuk menentukan prasyarat logis antarmateri.

TABLE I. DAFTAR MATERI DAN PRASYARATNYA

Label	Materi	Prasyarat
A	Kenal antar-anggota	-
B	Identitas Unit	-
C	Organogram Unit	B

D	Kedudukan dan Fungsi Unit	C
E	Posisi, Potensi, Peran Mahasiswa	G
F	Posisi, Potensi, Peran anggota unit	B, C, D, E, G
G	Tri Dharma Perguruan Tinggi	H
H	Definisi Pengabdian Masyarakat	-
I	Tujuan Pengabdian Masyarakat	H
J	Esensi dan Urgensi Pengabdian Masyarakat	G, H, I
K	Bentuk-bentuk Pengabdian Masyarakat	H
L	Realita Pengabdian Masyarakat	H, I, J
M	Hakikat Pendidikan di Indonesia	-
N	Kondisi Aktual Pendidikan di Indonesia	M
O	Empati	-
P	Komunikasi Sosial	O
Q	Komunikasi Efektif	O, P
R	Empathy Map Framework	O
S	Self Determination Theory	O
T	Experiential Learning Theory	O
U	Paradigma Pembelajaran	O, P
V	Kompetensi Pedagogik	O, S, T, U
W	Kompetensi Sosial	O, P, Q, R
X	Kompetensi Kepribadian	O
Y	Kompetensi Profesional	V
Z	Empati Guru	O
a	Definisi Masalah	-
b	Definisi Gejala Masalah	a
c	Why-why Diagram	a, b
d	MECE	a, b, c
e	How-how Diagram	a, b, d
f	RICE Framework	a, b, d
g	Critical Thinking	a, b, c, d
h	Creative Thinking	a, b, g, l
i	Heroic Leadership	g, h, j
j	SWOT & TOWS	-

k	Ingenuity	-
l	SCAMPER Framework	k
m	Simulasi Mengajar	V, W, X, Y, g, h

Tabel I menunjukkan daftar materi yang perlu disampaikan dilabeli dengan huruf besar dan kecil disertai prasyaratnya. Sebagai contoh, materi Organogram Unit dilabeli huruf C dan memiliki prasyarat materi berlabel B, yaitu Identitas Unit. Artinya, materi Identitas Unit harus disampaikan terlebih dahulu sebelum penyampaian materi organogram.

Relasi materi dan prasyarat pada Tabel I dimodelkan dalam bentuk himpunan pasangan berurutan pada kode Python berikut:

```
edges = [ # sisi graf dalam bentuk himpunan pasangan berurutan
("c", "d"), ("b", "c"), ("g", "e"), ("b", "f"), ("c", "f"), ("d", "f"), ("e", "f"), ("g", "f"),
("h", "g"), ("h", "i"), ("i", "j"), ("g", "j"), ("h", "j"), ("h", "k"), ("h", "l"), ("i", "l"),
("j", "l"), ("h", "n"), ("o", "p"), ("o", "q"), ("p", "q"), ("o", "r"), ("o", "s"), ("o", "t"),
("o", "u"), ("p", "u"), ("o", "v"), ("s", "v"), ("t", "v"), ("u", "v"), ("o", "w"), ("p", "w"),
("o", "x"), ("r", "x"), ("o", "y"), ("o", "z"), ("a", "b"), ("a", "c"), ("b", "c"),
("a", "d"), ("b", "d"), ("c", "d"), ("a", "e"), ("b", "e"), ("d", "e"), ("a", "f"), ("b", "f"),
("d", "f"), ("a", "g"), ("b", "g"), ("c", "g"), ("d", "g"), ("a", "h"), ("b", "h"), ("g", "h"),
("e", "i"), ("h", "i"), ("g", "j"), ("l", "h"), ("k", "l"), ("v", "m"), ("w", "m"),
("x", "m"), ("y", "m"), ("g", "m"), ("h", "m") ]
```

Gambar 3.1. Relasi materi dan prasyarat dalam himpunan pasangan berurutan di Python (sumber: dokumentasi pribadi)

Selanjutnya, hubungan materi dan prasyaratnya dimodelkan dalam bentuk graf berarah untuk merepresentasikan keterkaitan antarmateri. Setiap simpul dalam graf mewakili satu materi, sedangkan sisi antara simpul menunjukkan hubungan prasyarat. Materi u dengan prasyarat v dituliskan sebagai $u \rightarrow v$. Dengan menggunakan representasi ini, masalah penjadwalan dapat dimodelkan sebagai masalah penyortiran topologis.

B. Implementasi Algoritma

Algoritma Kahn diterapkan untuk menyelesaikan masalah pengurutan topologis. Implementasi algoritma dilakukan menggunakan pustaka Python, yaitu NetworkX, untuk mempermudah manipulasi graf. Berikut adalah langkah utama algoritma:

1. Hitung *in-degree* (jumlah sisi masuk) untuk setiap simpul.
2. Identifikasi simpul dengan *in-degree* nol yang dapat langsung dijadwalkan.
3. Iterasi melalui simpul-simpul tersebut, hapus simpul dari graf, dan perbarui *in-degree* simpul-simpul tetangganya.
4. Ulangi proses hingga semua simpul dijadwalkan atau hingga ditemukan siklus yang mengindikasikan konflik dependensi

Berikut adalah implementasi algoritma Kahn dalam bahasa Python:

```
import networkx as nx
def kahn_algorithm(graph):
    in_degree = {node: 0 for node in graph.nodes}
    for node in graph.nodes:
        for neighbor in graph.successors(node):
            in_degree[neighbor] += 1
    zero_in_degree = [node for node, degree in in_degree.items() if degree == 0]
    topo_order = []
    while zero_in_degree:
        current_node = zero_in_degree.pop(0)
        topo_order.append(current_node)
        for neighbor in graph.successors(current_node):
            in_degree[neighbor] -= 1
            if in_degree[neighbor] == 0:
                zero_in_degree.append(neighbor)
    return topo_order
```

Gambar 3.1. Implementasi algoritma Kahn dalam bahasa Python (sumber: dokumentasi pribadi)

C. Evaluasi Hasil

Evaluasi dilakukan untuk memastikan bahwa jadwal yang dihasilkan memenuhi setiap prasyarat. Evaluasi dilakukan dengan fokus pada pembuktian bahwa algoritma yang diusulkan menghasilkan jadwal yang logis, terstruktur, dan sesuai dengan aturan dependensi secara efisien.

IV. PEMBAHASAN

Pengujian dilakukan dengan menjalankan kode berikut:

```
edges = [ # sisi graf dalam bentuk himpunan pasangan berurutan...
G = nx.DiGraph()
G.add_edges_from(edges)
topological_order = kahn_algorithm(G)
label_to_name = { # map label dengan nama materi...
all_nodes = set(G.nodes)
sorted_nodes = set(topological_order)
unsorted_nodes = all_nodes - sorted_nodes
complete_order = topological_order + list(unsorted_nodes)
topological_order_names = [label_to_name[node] for node in complete_order]
i = 1
for x in topological_order_names:
    print(f"{i}. {x}")
    i += 1
```

Gambar 4.1. Implementasi proses penyortiran dalam bahasa Python (sumber: dokumentasi pribadi)

Setelah kode tersebut dijalankan, didapat hasil berikut:

```
1. Identitas Unit
2. Definisi Pengabdian Masyarakat
3. Hakikat Pendidikan di Indonesia
4. Empati
5. Definisi Masalah
6. Ingenuity
7. Organogram Unit
8. Tri Dharma Perguruan Tinggi
9. Tujuan Pengabdian Masyarakat
10. Bentuk-bentuk Pengabdian Masyarakat
11. Kondisi Aktual Pendidikan di Indonesia
12. Komunikasi Sosial
13. Empathy Map Framework
14. Self Determination Theory
15. Experiential Learning Theory
16. Kompetensi Kepribadian
17. Empati Guru
18. Definisi Gejala Masalah
19. SCAMPER Framework
20. Kedudukan dan Fungsi Unit
21. Posisi, Potensi, Peran Mahasiswa
22. Esensi dan Urgensi Pengabdian Masyarakat
23. Komunikasi Efektif
24. Paradigma Pembelajaran
25. Why-why Diagram
26. Posisi, Potensi, Peran anggota unit
27. Realita Pengabdian Masyarakat
28. Kompetensi Sosial
29. Kompetensi Pedagogik
30. MECE
31. Kompetensi Profesional
32. How-how Diagram
33. RICE Framework
34. Critical Thinking
35. Creative Thinking
36. SWOT & TOWS
37. Heroic Leadership
38. Simulasi Mengajar

[Done] exited with code=0 in 2.215 seconds
```

Gambar 4.2. Hasil proses penyortiran materi
(sumber: dokumentasi pribadi)

Hasil implementasi algoritma Kahn pada penjadwalan penyampaian materi kaderisasi menghasilkan urutan seperti pada Gambar 4.2. Dari hasil yang diperoleh, dapat terlihat bahwa algoritma Kahn berhasil menyusun jadwal dengan mempertimbangkan prasyarat antarmateri. Sebagai contoh:

- Materi “Identitas Unit” (1) ditempatkan sebelum “Organogram Unit” (20) dan “Kedudukan dan Fungsi Unit” yang memang membutuhkan pemahaman awal tentang identitas organisasi.
- “Definisi Pengabdian Masyarakat” (2) muncul sebelum “Tujuan Pengabdian Masyarakat” (9) dan “Esensi dan Urgensi Pengabdian Masyarakat” (22)
- “Empati” (4) dijadwalkan sebelum “Empathy Map Framework” (13)

Urutan ini menunjukkan bahwa algoritma telah bekerja sesuai ekspektasi, menghasilkan jadwal yang logis. Pendekatan algoritmik memberikan beberapa keunggulan signifikan dibandingkan metode manual. Pendekatan ini dapat menyelesaikan proses penjadwalan dalam waktu singkat, bahkan untuk graf dengan jumlah materi yang besar dan kompleksitas tinggi. Pada kasus ini, proses penyortiran selesai dalam 2.215 detik. Selain itu, hasil yang dihasilkan oleh algoritma selalu konsisten dan bebas dari kesalahan manusia yang sering terjadi dalam proses manual sehingga dapat dipastikan jadwal yang dihasilkan tidak melanggar aturan dependensi. Selain itu, algoritma dapat dengan mudah diterapkan pada skenario dengan jumlah materi yang lebih besar atau perubahan ketergantungan antarmateri.

Namun demikian, terdapat kelemahan yang muncul dari hasil implementasi algoritma Kahn, yaitu urutan materi yang terasa tersebar, terutama ketika terdapat kelompok materi yang seharusnya disampaikan secara berurutan. Contohnya adalah “Identitas Unit” (1), “Organogram Unit” (7), dan “Kedudukan

Fungsi Unit” (20). Ketiga materi ini memiliki keterkaitan yang erat dalam konteks pengenalan unit sehingga idealnya disampaikan secara berurutan dalam satu sesi. Namun, algoritma Kahn hanya mempertimbangkan hubungan logis berdasarkan graf dependensi tanpa memperhatikan konteks kesatuan topik.

Hasil ini menunjukkan bahwa algoritma Kahn memberikan urutan yang benar secara logis tetapi kurang aplikatif dalam beberapa kasus, karena tidak memperhitungkan aspek praktis pengelompokan materi yang serupa.

Untuk mengatasi kelemahan ini, solusi yang dapat diterapkan adalah pengelompokan materi ke dalam satu simpul atau grup berdasarkan kesamaan konteks. Dalam contoh di atas, materi “Identitas Unit”, “Organogram Unit”, dan “Kedudukan dan Fungsi Unit” dapat dikelompokkan ke dalam satu grup bernama “Pengenalan Unit”. Dengan pendekatan ini, ketiga materi tersebut dianggap sebagai satu kesatuan dalam graf dependensi, sehingga algoritma menghasilkan urutan yang lebih praktis. Selain itu, penyampaian materi menjadi lebih terstruktur dan logis dan mengikuti alur yang mudah dipahami oleh kader.

V. KESIMPULAN

Makalah ini menunjukkan bahwa algoritma Kahn merupakan metode yang efektif untuk menyusun jadwal penyampaian materi kaderisasi, terutama dalam memastikan bahwa prasyarat antarmateri terpenuhi. Hasil implementasi algoritma menunjukkan urutan penyampaian yang logis berdasarkan hubungan dependensi materi. Hal ini membuktikan bahwa algoritma Kahn mampu menyelesaikan masalah penjadwalan dengan pendekatan yang sistematis dan efisien.

Namun, terdapat kelemahan dalam penerapan algoritma ini, yaitu kurangnya perhatian terhadap pengelompokan materi yang serupa atau memiliki keterkaitan tema. Akibatnya, beberapa materi yang seharusnya disampaikan secara berurutan menjadi tersebar dalam jadwal. Sebagai solusi, pengelompokan materi berdasarkan kesamaan konteks dapat diterapkan untuk menghasilkan jadwal yang lebih praktis dan sesuai dengan kebutuhan kaderisasi.

Dengan menggabungkan algoritma Kahn dan pendekatan pengelompokan materi, diharapkan proses penyusunan jadwal dapat lebih optimal dan efisien, tidak hanya dalam memastikan urutan logis tetapi juga dalam meningkatkan keteraturan dan kemudahan pemahaman bagi peserta.

VI. LAMPIRAN

Implementasi kode dalam makalah ini dapat dilihat melalui GitHub:

<https://github.com/iammadfsq/Optimasi-Penjadwalan-Penyampaian-Materi-Kaderisasi>

Video penjelasan makalah dapat dilihat melalui YouTube:
<https://youtu.be/tDSPNTLaugU>

VII. UCAPAN TERIMA KASIH

Penulis ingin menyampaikan ucapan terima kasih kepada semua pihak yang telah memberikan dukungan hingga

makalah ini dapat terselesaikan dengan baik. Pertama-tama, penulis mengucapkan terima kasih kepada Allah SWT. atas nikmat dan karunianya. Penulis juga mengucapkan terima kasih kepada Pak Arrival Dwi Sentosa, S.Kom., M.T. dan Pak Dr. Ir. Rinaldi Munir, M.T. atas ilmu yang diberikan selama perkuliahan Matematika Diskrit.

REFERENSI

- [1] Tim Materi dan Metode Akademi Skhole 2024. (2024). *Buku Besar Akademi Skhole 2024*.
- [2] Munir, R. (2024). *Mata Kuliah Matematika Diskrit 2024-2025*. Diakses pada 1 Januari 2025, dari <https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/matdis24-25.htm>.
- [3] GeeksforGeeks. (2024). Kahn's algorithm for Topological Sorting. *GeeksforGeeks*. <https://www.geeksforgeeks.org/topological-sorting-indegree-based-solution/>

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, 1 Januari 2025



Ahmad Syafiq
1352315