

Penentuan Jadwal Putar Film di Bioskop dengan Algoritma A*

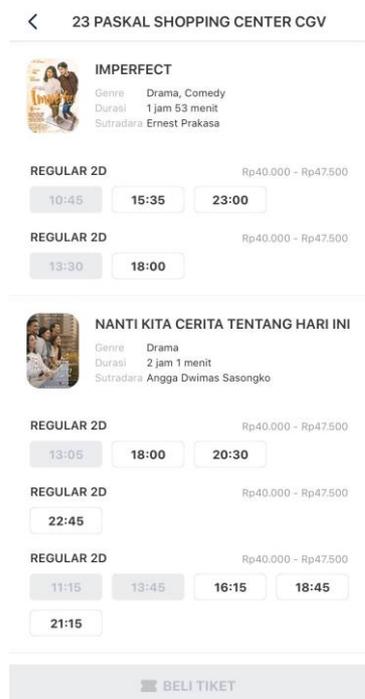
Mutiara Arifazzahra - 13518139

Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
13518139@std.stei.itb.ac.id

Abstract—Penentuan jadwal putar film adalah salah satu kasus penentuan keputusan di suatu bioskop. Dengan jadwal yang sesuai, maka suatu bioskop bisa menarik pelanggan secara maksimum sehingga dapat meraup keuntungan yang maksimum pula. Beberapa hal yang dipertimbangkan dalam menentukan jadwal tersebut adalah ekspektasi jumlah penonton suatu film, data penonton suatu film selama beberapa waktu terakhir (jika film tersebut sudah dalam masa tayang), dan kapasitas studio yang ada pada suatu bioskop. Dengan menggunakan salah satu algoritma optimasi, yakni A*, ditambah dengan beberapa data informasi yang tersedia (data yang didapatkan berdasarkan pengamatan maupun data heuristik), maka penentuan jadwal putar tersebut dapat dilakukan secara lebih sistematis (jika dibandingkan dengan cara manual).

Keywords—A* Algorithm, Scheduling Problem, Jadwal Tayang Film

I. PENDAHULUAN



Gambar 1.1. Contoh tampilan jadwal film pada salah satu aplikasi telepon genggam

Salah satu persoalan yang harus diputuskan pada suatu bioskop adalah penentuan jadwal putar film. Pihak yang berperan dalam memutuskan masalah tersebut disebut dengan *film programmer*. Bukan hanya mengatur jadwal putar film pada suatu bioskop, seorang *film programmer* juga berperan dalam pemilihan film yang dianggap layak untuk tayang, baik di bioskop, festival film, ataupun wadah penayangan lainnya.

Peran dari seorang *film programmer* pada dasarnya serupa dengan *TV programmer*. Keduanya sama-sama berperan untuk memilih tayangan yang paling cocok untuk tayang dan juga menjadwalkannya. Dalam melakukan penentuan jadwal tersebut, terdapat beberapa aspek yang harus diperhatikan. Jadwal tersebut sebaiknya memperhatikan ketertarikan para penonton. Hal ini dilakukan untuk memaksimalkan penonton untuk suatu tayangan tersebut sehingga keuntungan yang diperoleh juga dapat maksimal. Data ketertarikan tersebut dapat dilihat dari data penonton yang sudah terjadi pada beberapa waktu ke belakang ataupun melalui survey langsung [1].

Walaupun jarang terdengar namanya, seorang *film programmer* yang baik membutuhkan beberapa kemampuan [1]. Salah satu bekal utama yang dibutuhkan ialah ketertarikan dan pengetahuan terhadap dunia film, mulai dari keberagaman *genre* film hingga format film yang ada. Seorang *film programmer* juga harus memahami keinginan penonton. Pemahaman ini bisa didapat dari pengamatan terhadap suatu data atau melalui pengambilan data langsung, seperti melalui survei. Seorang *film programmer* yang baik juga memiliki penilaian/*judgement* yang baik terhadap suatu film. Mereka bisa melihat tayangan mana yang berpeluang menjadi terkenal dan menguntungkan. Di samping itu, kemampuan negosiasi yang baik juga dibutuhkan. Hal tersebut dapat membantu dalam berkomunikasi kepada peran lainnya yang ada dalam industri perfilman.

II. DASAR TEORI

A. Penentuan Jadwal Putar Film di Bioskop

Untuk melahirkan suatu jadwal putar film pada suatu bioskop, terdapat dua tahap yang harus dilalui. Tahap tersebut ialah pemilihan set film yang akan ditayangkan pada suatu bioskop dan dilanjut dengan penentuan jadwal putar film-film tersebut. Film yang akan ditayangkan dipilih berdasarkan beberapa pertimbangan, seperti ekspektasi jumlah penonton film tersebut. Penentuan jadwal film juga disusun sedemikian rupa

agar film yang dipilih tersebut dapat tayang dengan pengaturan tertentu yang dapat menghasilkan keuntungan terbesar.

Dalam menentukan jadwal putar film, dapat diamati data-data berikut:

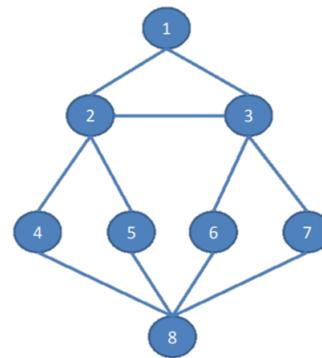
- Besar kapasitas penonton suatu studio pada suatu bioskop. Jika *demand* untuk suatu film bernilai tinggi, maka penayangan film tersebut dapat dilakukan di studio yang berkapasitas besar, atau jika hanya tersedia studio berkapasitas kecil, maka frekuensi penayangannya akan besar atau ditayangkan berkali-kali.
- Ekspektasi jumlah penonton suatu film untuk seminggu ke depan (jika film tersebut baru akan memulai masa tayang). Jika ekspektasi jumlah penonton suatu film bernilai besar, maka total durasi penayangan film dapat diperpanjang agar keuntungan yang diterima juga dapat lebih besar. Dengan menambah total durasi penayangan, berarti frekuensi penayangannya juga akan ditambahkan atau bernilai besar.
- Data rata-rata penonton suatu film dalam seminggu terakhir (jika film tersebut sudah dalam masa tayang). Jika nilai rata-rata penonton suatu film untuk suatu film tergolong tinggi, maka film tersebut tergolong populer dan memiliki kemungkinan untuk menarik penonton yang banyak pula. Jika jumlah penonton tinggi pada awal minggu sedangkan jumlahnya cenderung menurun pada akhir minggu, maka nilai rata-ratanya pun akan menurun. Hal ini berpengaruh pada perkiraan jumlah penonton untuk seminggu ke depannya.
- Permintaan/*demand* penonton terhadap suatu film pada waktu tersebut. *Demand* penonton dapat dilihat dengan kepopuleran suatu film tersebut, baik dari data penonton sebelumnya, popularitasnya di situs jejaring sosial atau situs lainnya, atau bisa juga dilihat dari jumlah frekuensi pencarian mengenai film tersebut di beberapa situs pencarian daring.
- Durasi pemutaran suatu film. Semakin lama durasi pemutaran suatu film, maka frekuensi penayangannya semakin kecil. Hal ini dikarenakan adanya batasan total durasi penayangan pada suatu bioskop. Jika *demand* untuk suatu film tergolong tinggi sedangkan durasinya tergolong panjang, maka hal ini dapat disiasati dengan memutarannya di beberapa studio atau pemutaran dilakukan selama beberapa hari hingga *demand* pelanggan tidak lagi tergolong tinggi (atau ada film lain yang *demand*-nya lebih tinggi).
- Total durasi pemutaran pada suatu studio/bioskop. Hal ini juga penting untuk diamati karena pemutaran film di suatu bioskop tidak bisa dilakukan secara terus menerus dikarenakan mayoritas bioskop tidak buka selama 24 jam, namun terdapat pula batasan jam operasional yang berlaku.

B. Algoritma Breadth First Search

Algoritma *breadth first search* (BFS) adalah salah satu algoritma pencarian yang bersifat *uninformed*, yakni pencarian yang tidak memiliki informasi tambahan mengenai perkiraan

ongkos yang diperlukan dari suatu simpul untuk menuju simpul tujuan/goal, atau bisa juga disebut bersifat buta. Ciri khas dari algoritma ini ialah teknis pembangkitan simpulnya dalam pengekspansian pohon pencarian saat bertambah tingkatannya. Algoritma ini juga harus memiliki ketentuan prioritas dalam pembangkitan simpulnya.

Pada saat akan melakukan ekspansi dari suatu simpul, semua simpul yang bertetangga dengan simpul tersebut akan dibangkitkan. Setelah semua simpul dibangkitkan, maka proses ekspansi akan berlanjut mulai dari simpul yang memiliki prioritas tertinggi terlebih dahulu. Setelah proses pembangkitan simpul tetangga selesai, maka seluruh simpul tetangga dari simpul yang memiliki prioritas setelah simpul tersebut juga dibangkitkan. Proses pembangkitan tersebut diulang hingga seluruh simpul yang berada dalam tingkat yang sama sudah dibangkitkan simpul tetangganya. Proses ini berulang terus hingga simpul *goal* sudah tercapai atau semua simpul sudah terbangkitkan.



Gambar 2.1. Contoh pohon pencarian [4]

Sebagai contoh, mengacu pada pohon pencarian pada gambar 2.1., jika penelusuran dimulai dengan simpul 1 sebagai simpul akar dengan angka yang lebih kecil memiliki prioritas yang lebih tinggi, maka urutan simpul yang akan dibangkitkan menurut algoritma BFS adalah $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8$.

Algoritma ini akan berjalan secara efektif jika keberadaan simpul *goal* terdapat pada tingkatan yang rendah, apalagi jika prioritasnya tinggi, sehingga simpul *goal* bisa segera dicapai.

C. Algoritma Uniform Cost Search

Algoritma *Uniform Cost Search* (UCS) adalah salah satu algoritma pencarian yang bersifat *uninformed*. Algoritma ini juga merupakan salah satu algoritma optimasi. Algoritma ini menganut dasar kerja menyerupai algoritma BFS dengan memprioritaskan ekspansi dari simpul yang memiliki ongkos paling optimal.

Cara kerja dari algoritma ini dimulai dengan pembangkitan semua tetangga dari suatu simpul yang diikuti dengan perhitungan ongkos untuk tiap simpulnya (jika dihitung dari simpul akar). Perhitungan ongkos ini dirumuskan dalam suatu fungsi objektif yang dapat dinotasikan sebagai fungsi $g(n)$. Ekspansi dilanjutkan pada satu simpul yang memiliki ongkos paling optimal. Simpul yang turut dipertimbangkan untuk diekspan bukan hanya simpul yang berada pada tingkatan yang

sama, tetapi juga simpul pada tingkat manapun, baik lebih rendah maupun lebih tinggi dari simpul terakhir yang diekspan. Proses ekspansi ini diulangi terus menerus (walaupun simpul *goal* sudah tercapai) hingga simpul hidup yang memiliki prioritas tertinggi ialah simpul *goal* (yang menandakan bahwa *goal* sudah tercapai dengan ongkos yang diperlukan sudah bernilai optimal).

D. Algoritma Greedy Best First Search

Algoritma *greedy best first search* (GBFS) adalah salah satu algoritma pencarian yang bersifat *informed*, yakni pencarian dengan informasi tambahan berbasis heuristik. Kelebihan dari algoritma yang bersifat *informed* adalah kemampuannya untuk mengetahui simpul mana yang lebih menjanjikan (dibandingkan simpul-simpul lainnya) untuk mencapai simpul *goal* dengan ongkos yang paling optimal. Informasi heuristik tersebut dapat dimanfaatkan dalam perhitungan ongkos tiap simpul sehingga dapat diperkirakan simpul mana yang paling optimal.

Dasar kerja dari algoritma GBFS menyerupai algoritma UCS dalam hal penentuan nilai ongkos tiap simpul dengan perumusan suatu fungsi objektif. Namun, pada algoritma ini, nilai fungsi objektif tersebut mengacu pada nilai heuristik yang dimiliki. Fungsi objektif dari algoritma GBFS bisa dinotasikan sebagai fungsi $h(n)$. Fungsi objektif ini menampilkan perkiraan sisa "jarak" yang harus ditempuh untuk mencapai simpul akar dari suatu simpul.

Seperti namanya, cara kerja dari algoritma ini juga terinspirasi dari algoritma *greedy* yang cenderung memilih suatu subsolusi yang dinilai paling menguntungkan, atau dalam kasus ini, optimal. Cara kerja dari algoritma GBFS sama seperti algoritma UCS, namun perhitungan nilai fungsi objektifnya saja yang berbeda.

Kekurangan dari algoritma ini ialah kemungkinan tidak tercapainya simpul *goal* jika nilai heuristik yang dimiliki tidak terlalu tepat (atau informasi yang diberikan tidak terlalu lengkap). Proses pencarian juga tidak bisa berjalan mundur/*backtrack* karena nilai heuristiknya pasti akan lebih tidak optimal.

E. Algoritma A*

Algoritma A* (dibaca: algoritma *A star*) adalah salah satu algoritma pencarian yang bersifat *informed*, yakni pencarian dengan informasi tambahan berbasis heuristik. Algoritma ini juga merupakan algoritma optimasi. Dalam algoritma A*, optimasi dilakukan untuk mengoptimalkan ongkos dengan tetap memperhatikan batasan/*constraint* dari suatu permasalahan.

Dasar kerja dari algoritma ini merupakan gabungan dari dua algoritma, yakni algoritma *uniform cost search* (UCS) dan *greedy best first search* (GBFS). Dalam pembangkitan simpul, prinsip yang digunakan menyerupai prinsip yang dianut algoritma BFS, yakni semua anak dari suatu simpul turut dibangkitkan (jika tidak melanggar *constraint* yang berlaku). Simpul yang selanjutnya diekspan adalah satu simpul dengan nilai objektif yang paling optimal. Cara kerja pembentukan pohon pencarian algoritma ini menyerupai algoritma UCS dan GBFS.

Fungsi objektif dari algoritma A* ini dirumuskan dalam

$$f(n) = g(n) + h(n)$$

dengan keterangan:

- $f(n)$: nilai fungsi objektif suatu simpul
- $g(n)$: nilai ongkos untuk mencapai simpul n dari simpul akar
- $h(n)$: nilai heuristik (atau nilai ongkos untuk mencapai simpul tujuan (*goal*) dari simpul n).

Nilai $g(n)$ pada algoritma ini menyerupai fungsi objektif dari algoritma UCS. Nilai $h(n)$ pada algoritma ini menyerupai fungsi objektif dari algoritma GBFS.

Nilai heuristik yang dimiliki suatu simpul dapat menunjukkan kerumitan dalam penyelesaian subprogram tersebut (jika mencantumkan simpul tersebut sebagai salah satu solusinya), kualitas solusi yang direpresentasikan simpul tersebut, ataupun banyak informasi yang dimiliki. Dalam menentukan nilai heuristik, dibutuhkan evaluasi untuk menentukan apakah nilai tersebut bersifat pantas untuk diterima (*admissible*) atau tidak. Suatu nilai bersifat *admissible* jika nilainya tidak melebihi nilai ongkos sebenarnya untuk mencapai simpul *goal*. Sebagai contoh, jika sedang menghitung nilai heuristik dalam jarak untuk mencapai suatu kota, nilai tersebut tidak boleh melebihi nilai jarak yang sebenarnya. Syarat kepastian nilai heuristik tersebut dapat dirumuskan sebagai

$$h(n) \leq h^*(n)$$

dengan keterangan:

- $h(n)$: nilai heuristik ongkos untuk mencapai simpul *goal* dari simpul n
- $h^*(n)$: nilai ongkos sebenarnya untuk mencapai simpul *goal* dari simpul n

Setelah simpul *goal* tercapai, semua simpul hidup yang memiliki nilai ongkos lebih besar dari *goal* dimatikan. Jika masih ada simpul hidup, pencarian dapat dilanjutkan. Hal ini mungkin saja terjadi dan menandakan bahwa terdapat beberapa opsi solusi dengan ongkos yang sama-sama optimal.

Algoritma A* bersifat komplet (simpul *goal* pasti tercapai/pasti ada penyelesaian). Algoritma ini juga akan memberikan solusi yang optimal asalkan nilai heuristik yang dipilih bersifat *admissible*. Kompleksitas yang dimiliki algoritma ini adalah $O(b^m)$ baik untuk kompleksitas waktu maupun ruang, dengan b adalah banyaknya simpul anak yang mungkin diekspan pada suatu simpul dan m adalah banyaknya tingkat suatu pohon pencarian.

III. IMPLEMENTASI

Untuk memecahkan suatu persoalan optimasi dengan algoritma A*, harus ditentukan terlebih dahulu perumusan untuk perhitungan fungsi objektif. Penentuan rumus tersebut dapat dibagi menjadi dua tahap, yaitu nilai dari $g(n)$ dan $h(n)$. Perumusan didapatkan dengan mempertimbangkan data-data yang harus diamati dalam menentukan jadwal putar film.

A. Nilai $g(n)$

Untuk menghitung nilai ongkos suatu simpul n dari simpul akar, dapat digunakan data perkiraan jumlah penonton yang dapat dirumuskan dengan

$$g(n) = audience(n) * demand(n)$$

dengan keterangan:

- n : pilihan film
- $g(n)$: nilai ongkos untuk mencapai simpul n dari simpul akar
- $audience(n)$: perkiraan jumlah penonton untuk suatu pilihan film n
- $demand(n)$: permintaan/*demand* penonton untuk suatu pilihan film n dalam saat waktu penentuan jadwal yang berlaku untuk suatu jangka waktu tertentu

yang didasari dengan perhitungan

$$audience(n) = \frac{expected(n) + lastweek(n)}{a}$$

dengan keterangan:

- n : pilihan film
- $audience(n)$: perkiraan jumlah penonton untuk suatu pilihan film n
- $expected(n)$: ekspektasi jumlah penonton untuk suatu pilihan film n
- $lastweek(n)$: jumlah penonton untuk suatu pilihan film n selama seminggu terakhir
- a : nilai pembagi berdasarkan banyak data yang tersedia. Bernilai 2 jika tersedia data $lastweek(n)$ dan bernilai 1 jika data tidak tersedia.

B. Nilai $h(n)$

Untuk menghitung nilai ongkos suatu simpul n dari simpul akar, dapat digunakan data perkiraan jumlah penonton yang dapat dirumuskan dengan

$$h(0) = totalduration - duration(n)$$

untuk simpul akar/film pertama, dengan keterangan:

- n : pilihan film
- $h(0)$: nilai heuristik simpul akar
- $duration(n)$: durasi pemutaran suatu pilihan film n , termasuk waktu pembersihan ruang studio (sebagai jeda transisi ke film selanjutnya)
- $totalduration$: total durasi pemutaran film pada suatu studio bioskop

dan bernilai

$$h(n) = h(n - 1) - duration(n)$$

untuk simpul akar/film pertama, dengan keterangan:

- n : pilihan film
- $h(n)$: nilai heuristik simpul n
- $h(n - 1)$: nilai heuristik simpul sebelum n
- $duration(n)$: durasi pemutaran suatu pilihan film n , termasuk waktu pembersihan ruang studio (sebagai jeda transisi ke film selanjutnya)

Nilai heuristik ini termasuk ke dalam kategori *admissible* dikarenakan durasi suatu pilihan film sudah menghitung jeda waktu untuk pembersihan ruang studio, yang sesuai dengan kasus pada dunia nyata.

C. Pembangkitan Simpul dan Pembentukan Pohon Pencarian

Dalam pohon pencarian algoritma ini, hasil dari satu pohon pencarian merepresentasikan urutan film yang akan diputar pada suatu studio di bioskop dalam suatu hari. Suatu simpul n merepresentasikan pilihan film yang akan diputar. Dalam menentukan simpul yang akan dibangkitkan, maka simpul-simpul yang akan dibangkitkan tersebut harus memenuhi dua syarat, yakni:

1) *Perhitungan kapasitas studio*: Dihitung terlebih dahulu apakah nilai $audience(n)$ dari suatu simpul n tersebut cukup untuk ditampung pada suatu studio. Hal tersebut dirumuskan dalam

$$audience(n) \leq capacity(k)$$

dengan keterangan:

- $audience(n)$: perkiraan jumlah penonton untuk suatu pilihan film n
- $capacity(k)$: kapasitas penonton yang dapat ditampung dalam suatu studio k

2) *Perhitungan sisa durasi*: Durasi dari film yang dipilih tidak bisa melebihi sisa durasi yang tersedia dalam suatu studio. Hal ini dirumuskan dalam

$$duration(n) \leq h(n - 1)$$

dengan keterangan:

- $duration(n)$: durasi pemutaran suatu pilihan film n , termasuk waktu pembersihan ruang studio (sebagai jeda transisi ke film selanjutnya)
- $h(n - 1)$: nilai heuristik simpul sebelum n

Jika kedua syarat tersebut dipenuhi oleh suatu simpul n , maka simpul tersebut boleh dibangkitkan. Jika sudah tidak ada lagi simpul yang memenuhi minimal salah satu dari syarat tersebut, maka sudah tidak ada lagi film yang bisa diputar dalam sisa jangka waktu yang tersedia.

Karena persoalan penentuan jadwal putar film ini merupakan persoalan optimasi, maka simpul yang akan lanjut dieskpan juga harus mempunyai nilai fungsi objektif yang optimal, atau dalam kasus ini, maksimal. Pada awal pembentukan pohon pencarian, simpul yang dibangkitkan hanyalah satu, yakni untuk menjadi simpul akar. Simpul tersebut juga merupakan simpul yang memiliki nilai fungsi objektif terbesar di antara pilihan simpul

lainnya. Selain untuk pembangkitan simpul akar, maka semua simpul yang memenuhi syarat akan dibangkitkan, walaupun yang akan lanjut diekspansi hanya salah satu simpul dengan nilai objektif terbesar. Pembangkitan simpul terus dijalankan hingga simpul *goal* tercapai. Dalam hal ini, simpul *goal* tersebut ialah simpul terakhir yang memenuhi kedua syarat pembangkitan simpul.

IV. PENGUJIAN

Pada bagian ini, akan dilakukan pengujian algoritma terhadap suatu data *dummy* yang mencerminkan data set film yang dijadwalkan untuk tayang di suatu bioskop untuk seminggu ke depan. Persoalan yang harus diselesaikan adalah penentuan jadwal putar film untuk salah satu studio pada bioskop tersebut untuk satu hari (untuk menentukan jadwal putar untuk jangka waktu seminggu ke depan, maka proses pencarian diulang sebanyak hari penayangan).

Diketahui terdapat studio A pada bioskop A-Cinema memiliki kapasitas maksimum 80 orang. Terdapat beberapa film yang akan diputar sebagai berikut

TABLE I. DAFTAR FILM YANG AKAN DIPUTAR

Judul Film	Data Film	
	Keterangan	Nilai
Alva	Perkiraan jumlah penonton	50 orang
	Rata-rata penonton perhari selama seminggu terakhir	66 orang
	<i>Demand</i>	50%
	Durasi film	2 jam
Brafo	Perkiraan jumlah penonton	60 orang
	<i>Demand</i>	60%
	Durasi film	3 jam
Charly	Perkiraan jumlah penonton	30 orang
	Rata-rata penonton perhari selama seminggu terakhir	70 orang
	<i>Demand</i>	30%
	Durasi film	1 jam
Detta	Perkiraan jumlah penonton	100 orang
	Rata-rata penonton perhari selama seminggu terakhir	150 orang
	<i>Demand</i>	70%
	Durasi film	2 jam

^a *Demand* berupa persentase dengan keterangan semakin besar nilainya maka *demand*-nya semakin tinggi pula.

^b Setelah diberikan satu jadwal penayangan, maka nilai *demand* berkurang 10% untuk perhitungan pada hari yang sama. Untuk perhitungan pada hari yang berlainan, nilai *demand* akan dihitung mulai dari data yang tertera.

^c Durasi film sudah termasuk waktu transisi antarfilm (untuk keperluan pembersihan dan perapihan studio).

Karena kapasitas studio adalah 80 orang, film Detta tidak memungkinkan untuk ditayangkan di studio ini. Urutan pembangkitan simpul pada tiap tingkatannya digambarkan pada tabel-tabel berikut ini.

Film yang akan ditayangkan pertama kali ditentukan pada pemilihan simpul akar. Menurut "Tabel II", film yang dipilih adalah film Brafo.

TABLE II. PERHITUNGAN FUNGSI OBJEKTIF TINGKAT KE-0

Nilai Fungsi	Pilihan Film		
	Alva	Brafo	Charly
$g(n)$	29	36	15
$h(n)$	10	9	11
$f(n)$	39	45	26

Film yang akan ditayangkan pertama kali ditentukan pada pemilihan simpul pada tingkat ke-1. Menurut "Tabel III", film dengan urutan penayangan ke-2 adalah film Brafo.

TABLE III. PERHITUNGAN FUNGSI OBJEKTIF TINGKAT KE-1

Nilai Fungsi	Pilihan Film		
	Alva	Brafo	Charly
$g(n)$	29	30	15
$h(n)$	7	6	8
$f(n)$	36	36	23

^a Nilai $f(n)$ film Alva dan Brafo sama. Film dengan jumlah audience (n) yang lebih besar diprioritaskan untuk mengoptimalkan keuntungan yang didapat.

Film yang akan ditayangkan pertama kali ditentukan pada pemilihan simpul pada tingkat ke-2. Menurut "Tabel IV", film dengan urutan penayangan ke-3 adalah film Alva.

TABLE IV. PERHITUNGAN FUNGSI OBJEKTIF TINGKAT KE-2

Nilai Fungsi	Pilihan Film		
	Alva	Brafo	Charly
$g(n)$	29	24	15
$h(n)$	4	3	5
$f(n)$	33	27	20

Film yang akan ditayangkan pertama kali ditentukan pada pemilihan simpul pada tingkat ke-3. Menurut "Tabel V", film dengan urutan penayangan ke-4 adalah film Alva.

TABLE V. PERHITUNGAN FUNGSI OBJEKTIF TINGKAT KE-3

Nilai Fungsi	Pilihan Film		
	Alva	Brafo	Charly
$g(n)$	23.2	24	15
$h(n)$	2	1	3
$f(n)$	25.2	25	18

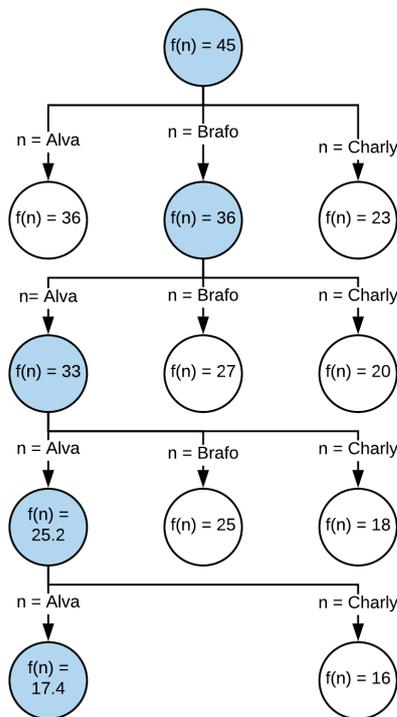
Film yang akan ditayangkan pertama kali ditentukan pada pemilihan simpul pada tingkat ke-4. Menurut "Tabel VI", film dengan urutan penayangan ke-5 adalah film Alva.

TABLE VI. PERHITUNGAN FUNGSI OBJEKTIF TINGKAT KE-4

Nilai Fungsi	Pilihan Film	
	Alva	Charly
$g(n)$	17.4	15
$h(n)$	0	1
$f(n)$	17.4	16

^a. Film Brafo tidak lagi memenuhi syarat $duration(n) \leq h(n-1)$ yang menandakan sisa durasi penayangan tidak cukup untuk menayangkan film tersebut.

Dari tabel-tabel di atas, dapat digambarkan pohon pencarian sebagai berikut.



Gambar 4.1. Pohon pencarian penentuan jadwal putar film pada studio A di bioskop A-Cinema

Dari gambar 4.1., didapatkan bahwa salah satu solusi urutan untuk penentuan jadwal putar film ialah Film Brafo → Brafo → Alva → Alva → Alva dengan total perkiraan penonton 294.

V. KESIMPULAN

Untuk memecahkan persoalan penentuan jadwal putar film, algoritma A* dapat diimplementasikan dan solusi yang dihasilkan juga bisa dinilai optimal. Namun, dikarenakan algoritma A* merupakan algoritma yang bersifat *informed*, diperlukan konsiderasi mengenai data-data yang apa saja yang tepat dan sesuai. Ketepatan informasi dan penentuan rumus fungsi objektif juga mempengaruhi kualitas dari solusi tersebut.

VIDEO LINK DI YOUTUBE

Video penjelasan lisan dari makalah ini dapat dilihat pada laman berikut

<https://youtu.be/XLfn2alUdWA>

ACKNOWLEDGMENT

Penulis mengucapkan puji syukur kepada Allah SWT karena berkat kehendak-Nya penulis dapat menyelesaikan tugas terakhir untuk mata kuliah Strategi Algoritma ini. Penulis juga berterima kasih kepada seluruh pihak yang telah terlibat dalam penulisan makalah ini, baik dari keluarga penulis yang turut membantu menyediakan bekal energi berbentuk makanan untuk penulis, hingga teman-teman mahasiswa lain yang turut menemani dan *men-support* penulis dalam penyusunan makalah ini. Tak lupa, terima kasih juga penulis sampaikan kepada tim dosen serta asisten mata kuliah Strategi Algoritma yang telah menuntun penulis dalam keseharian pembelajaran mata kuliah ini. Walaupun kegiatan pembelajaran di kelas tidak dapat dilaksanakan, namun tim pengajar telah mengerahkan usaha terbaiknya untuk tetap memberi pembelajaran kepada kami, paramahasiswa. Terakhir, penulis juga sangat berterima kasih kepada para pemilik referensi yang penulis gunakan pada penyusunan makalah ini karena penulis mendapatkan banyak pengetahuan baru selama penulis menyusun makalah ini.

Penulis berharap makalah ini dapat bermanfaat di kemudian hari. Semoga proses pengolahan data yang dapat digunakan sebagai informasi heuristik bisa lebih akurat lagi sehingga algoritma yang bersifat *informed* dapat meningkatkan kualitas solusinya. Selain itu, semoga perkembangan algoritma untuk penyelesaian persoalan, terutama persoalan optimasi, juga bisa lebih pesat lagi sehingga persoalan yang biasa dihadapi dalam kehidupan sehari-hari dapat lebih mudah diselesaikan.

REFERENSI

- [1] ScreenSkills, "Film Programmer" [Online] <https://www.screenkills.com/careers/job-profiles/film-and-tv-drama/sales-and-distribution/film-programmer/> (Diakses 3 Mei 2020)
- [2] T. Dabbas, N. Guran, dan A. Tascioglu, "Scheduling of Movies at Lincoln Plaza Cinema Theaters" [Online] <http://www.columbia.edu/~cs2035/courses/ieor4405.S13/p4.pdf> (Diakses 2 Mei 2020)
- [3] J. Eliashberg dkk., "Demand-driven Scheduling of Movies in a Multiplex" [Online] <https://faculty.wharton.upenn.edu/wp-content/uploads/2012/04/Demand-driven-scheduling-of-movies-in-a-multiplex.pdf> (Diakses 2 Mei 2020)
- [4] Rinaldi Munir, "Breadth First Search (BFS) dan Depth First Search (DFS)" [Online] [http://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2019-2020/BFS-dan-DFS-\(2020\).pdf](http://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2019-2020/BFS-dan-DFS-(2020).pdf) (Diakses 3 Mei 2020)
- [5] Rinaldi Munir, "Route-Planning Using A Star and UCS" [Online] [http://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2017-2018/A-Star-Best-FS-dan-UCS-\(2018\).pdf](http://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2017-2018/A-Star-Best-FS-dan-UCS-(2018).pdf) (Diakses 3 Mei 2020)

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 Mei 2020

Mutiara Arifazzahra - 13518139