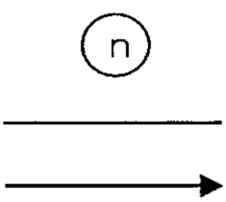
BAB II

MATERI PENUNJANG

2.1 Pengertian dasar graph

Permasalahan rute terpendek pada jaringan jalan dapat dimodelkan kedalam bentuk jaringan yang berupa graph berarah $G(N, A)$. Tempat atau terminal diwakili oleh suatu *node* n sedangkan jalan yang dilalui direpresentasikan dalam bentuk garis berarah atau *arc* a. Di bawah ini diberikan beberapa definisi tentang graph, dan pengertian mengenai makna gambar-gambar:



: lingkaran menggambarkan *node*

: garis mengambarkan *edge*

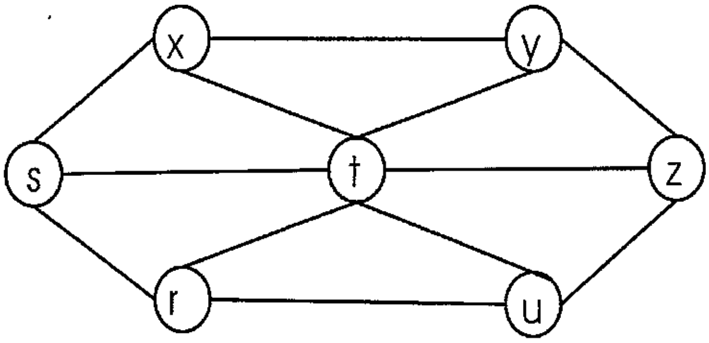
: garis berarah mengambarkan *arc*

Definisi 2.1

Suatu graph yang dinotasikan dengan $G(N, E)$ adalah himpunan berhingga titik (*node*) $N(G)$, yang tidak kosong dan himpunan garis (*edge*) $E(G)$, yang mungkin kosong.

(Chartrand,G. dan O.R.Oellerman., 1993)

Berikut ini diberikan contoh graph dengan 7 *node* dan 12 *edge*.



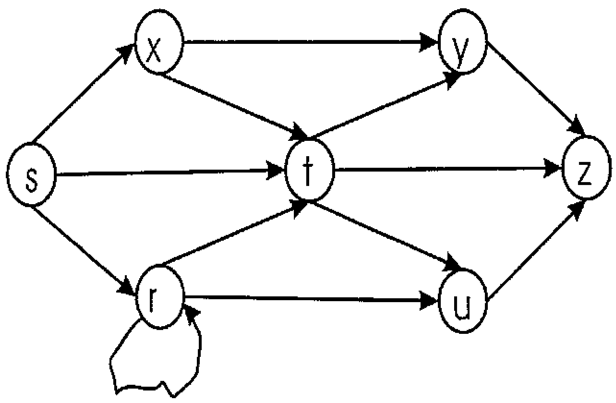
Gambar 2.1
Graph dengan 7 *node* dan 12 *edge.*

Definisi 2.2

Sebuah graph berarah atau *directed* graph $G(N, A)$ terdiri dari himpunan berhingga *node n*, $N(G)$ dan himpunan *arc a*, $A(G)$ yang elemennya merupakan pasangan berurutan dari *node* yang berbeda yaitu $a = (i, j)$, dimana *i* dan *j* adalah *node* yang berbeda.

(Chartrand,G. dan O.R.Oellerman., 1993)

Jumlah *node* dalam graph menyatakan *order*, sedangkan jumlah *arc* menyatakan *size*. Sebagai ilustrasi, diberikan contoh graph berarah:



Gambar 2.2
Graph berarah

Dari Gambar 2.2 dapat dijelaskan bahwa graph tersebut mempunyai *order* 7, dan *size* 13.

Definisi 2.3

Sebuah *arc* (i, j) adalah sebuah garis berarah yang menghubungkan *node* awal *i dan node* akhir *j*. (Chartrand,G. dan O.R.Oellerman., 1993)

Jika *arc* $(i, j) \in A(G)$ maka *node* *i* dikatakan *adjacent* ke *j*, dan *node j* dikatakan *adjacent* dari *i*. *Arc* (i, j) *incident* dengan *node i* dan *node j*. Sebuah garis (i, j) yang mempunyai *node* awal sama dengan *node* akhir disebut *loop*.

Sebagai gambaran, perhatikan *arc* (s, x) pada Gambar 2.2, yang menggambarkan *node s adjacent* ke *x* dan *node x adjacent* dari *s*, serta *arc* (s, x) *incident* dengan *node s* dan *x*. Graph berarah pada Gambar 2.2 mempunyai satu *loop*, yaitu *arc* (r, r).

Definisi 2.4

Suatu graph berarah yang setiap *node* atau *arc* atau keduanya berbobot disebut graph berbobot (*weight graph*), dinotasikan dengan $G(N, A, l)$. (Chartrand,G. dan O.R.Oellerman., 1993)

Hal ini dapat dilihat pada fenomena jaringan jalan raya yang menggunakan lampu lalu-lintas. Jaringan tersebut mempunyai waktu tunggu pada setiap *node*, dengan waktu tunggu diatur oleh lampu lalu-lintas. Setiap jalan yang menghubungkan persimpangan jalan, mempunyai waktu perjalanan yang didefinisikan sebagai waktu yang diperlukan untuk melakukan perjalanan dalam jalan tersebut. Jaringan ini merupakan salah satu contoh graph berbobot, dengan waktu tunggu pada persimpangan jalan merupakan bobot *node*, dan waktu perjalanan adalah bobot *arc*.

Misalkan diketahui data waktu tunggu tiap persimpangan jalan dan waktu perjalanan untuk tiap jalan yang diberikan dalam bentuk tabel sebagai berikut:

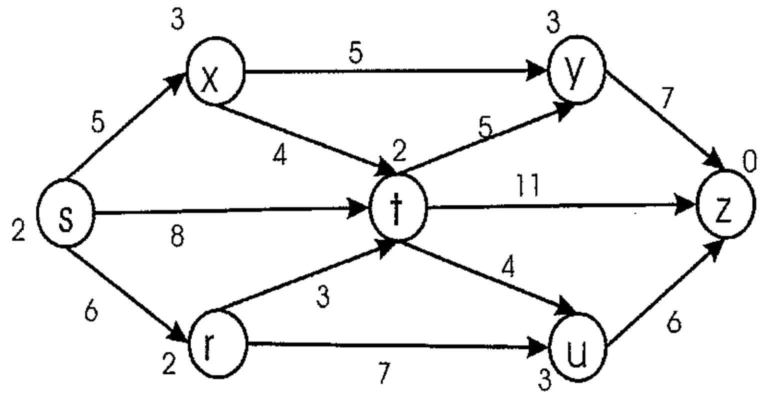
Tabel 2.1 Data waktu perjalanan pada jaringan jalan

No	Jalan raya	Waktu perjalanan (menit)
1	$s-x$	5
2	$s-t$	8
3	$s-r$	6
4	$x-y$	5
5	$x-t$	4
6	$r-t$	3
7	$r-u$	7
8	$t-y$	5
9	$t-z$	11
10	$t-u$	4
11	$y-z$	7
12	$u-z$	6

Tabel 2.2 Data waktu tunggu pada persimpangan jalan

No	Persimpangan jalan	Waktu tunggu (menit)
1	s	2
2	x	3
3	t	2
4	r	2
5	y	3
6	u	3
7	z	0

Dari tabel di atas dapat dibuat sebuah graph berbobot yang merepresentasikan suatu jaringan jalan yang menggunakan lampu lalu-lintas, dengan bobot *node* adalah waktu tunggu pada tiap-tiap persimpangan jalan, dan bobot *arc* adalah waktu perjalanan untuk jalan yang menghubungkan tiap persimpangan jalan. Graph berbobot tersebut dapat digambarkan sebagai berikut:



Gambar 2.3
Graph berbobot

Definisi 2.5

Bobot *node* yang dinotasikan dengan $l(n)$ atau $l(i)$ adalah nilai atau harga dari suatu *node* dalam graph.

(Chartrand,G. dan O.R.Oellerman., 1993)

Bobot *node* dapat bernilai positif, negatif, atau nol.

Contoh 2.1

Lihat graph berbobot pada Gambar 2.3.

$l(s) = 2$, artinya bobot *node s* sama dengan 2.

$l(x) = 3$, artinya bobot *node x* sama dengan 2.

$l(t) = 2$, artinya bobot *node t* sama dengan 2.

Definisi 2.6

Bobot garis yang dinotasikan dengan $l(e)$ atau $l(i,j)$ adalah nilai atau harga dari suatu garis dalam graph.

(Chartrand,G. dan O.R.Oellerman., 1993)

Bobot garis dapat bernilai positif, negatif, atau nol. Untuk memperjelas definisi di atas perhatikan contoh berikut ini:

Contoh 2.2

Perhatikan graph berbobot pada Gambar 2.3.

$l(s,x) = 5$, artinya bobot *arc* (s,x) sama dengan 5.

$l(s,t) = 8$, artinya bobot *arc* (s,t) sama dengan 8.

Dalam jaringan jalan menggunakan lampu lalu-lintas, bobot suatu *node* dan garis dapat berupa jarak, waktu, atau biaya untuk *node* atau garis tersebut.

2.2 *Walk* dan *cycle*

2.2.1 *Walk*

Pada permasalahan rute terpendek dalam jaringan jalan yang menggunakan lampu lalu-lintas, setiap rute perjalanan pada jaringan tersebut dapat digambarkan kedalam *path* atau lintasan pada graph yang menghubungkan *node* awal dan *node* tujuan.

Definisi 2.7

Walk adalah deretan bergantian antara titik (*node*) dan garis (*edge*) yang dimulai dan diakhiri dengan titik,

$W: n_0, e_1, n_1, e_2, n_2, e_3, .., n_{n-1}, e_n, n_n \quad (n \geq 0)$,

dengan $e_i = n_{i-1}n_i$, untuk $i = 1,2,3,...,n$.

(Chartrand,G. dan O.R.Oellerman., 1993)

Trail adalah *walk* dimana tidak ada garis yang diulang, sedangkan *path* adalah *walk* dengan tidak ada titik yang diulang. Untuk memudahkan penulisan *walk*, maka penulisan garis pada *walk* tersebut dapat dihilangkan. Oleh karena itu barisan titik dan garis dalam *walk* W di atas dapat dituliskan sebagai $W: n_0, n_1, n_2, n_3, ..., n_n$.

Dari graph pada Gambar 2.1 (hal. 5), yang disebut *walk* adalah W: $s, x, y, t, r, u, t, y, z$ sedangkan *trail* adalah W: s, r, t, x, y, t, u, z, dan *path* adalah W: s, t, r, u, z. Semua pengertian mengenai *walk* pada Definisi 2.7 juga berlaku untuk graph berarah.

Di dalam suatu *walk* $n - m$ pada graph G, terdapat sebuah *path* $n - m$ yang menghubungkan *node* n dan m. Jadi *path* dalam *walk* tersebut adalah tunggal.

Teorema 2.1

Setiap *walk* $n - m$ dalam sebuah graph mempunyai sebuah *path* $n - m$.

Bukti

Misal W adalah *walk* $n - m$ pada sebuah graph G. Jika $n = m$, maka terdapat jawaban trivial yaitu *path* $n - n$.

Diasumsikan bahwa $n \neq m$, andaikan $W : n_0 = n, n_1, n_2, ..., n_n = m$. Jika tidak ada *node* dari G yang muncul lebih dari satu kali dalam *walk* W, maka W itu sendiri adalah *path* $n - m$.

Dilain pihak jika terdapat beberapa *node* dalam G yang muncul lebih dari satu kali dalam *walk*. Misalkan i dan j adalah bilangan integer positif yang berbeda dengan i < j, sedemikian sehingga $n_i = n_j$. Jika hubungan $n_i, n_{i+1}, ..., n_{j-1}$ dihilangkan dari W, maka diperoleh sebuah *walk* $n - m$, sebut W_1 yang mempunyai panjang kurang dari W. Jika tidak ada *node* pada G yang muncul lebih dari satu kali dalam *walk* W_1, maka W_1 adalah *path* $n - m$.

Jika belum ditemukan kondisi seperti ini, maka kembali ke prosedur di atas, sampai akhirnya diperoleh *path* $n - m$.

Banyak rute perjalanan yang dapat digunakan untuk mencapai tempat tujuan pada jaringan jalan. Rute tersebut mempunyai waktu perjalanan yang berbeda-beda. Dari rute perjalanan yang ada dicari rute perjalanan terpendek, yaitu rute yang mempunyai waktu perjalanan minimal. Dengan kata lain, lintasan tersebut merupakan lintasan terpendek yang menghubungkan *node* awal ke *node* tujuan. Lintasan ini mempunyai bobot lintasan minimal.

Definisi 2.8

Bobot lintasan dari *node* i ke *node* j, yang dilambangkan dengan $P(i,j)$ adalah jumlah semua bobot *node* dan *arc* yang membentuk lintasan tersebut. (Chartrand,G. dan O.R.Oellerman., 1993)

Contoh 2.3

Perhatikan lintasan $W: s, x, y, z$ yang menghubungkan *node* s dan z pada Gambar 2.3 (hal. 8). Lintasan ini terdiri dari 3 *arc* dan 4 *node*. Berikut ini diberikan bobot masing-masing *node* dan *arc* yang menyusun lintasan tersebut:

Bobot *node* $s = l(s) = 2$

Bobot *node* $x = l(x) = 3$

Bobot *node* $y = l(y) = 3$

Bobot *node* $z = l(z) = 0$

Bobot garis $s - x = l(s,x) = 5$

Bobot garis $x - y = l(x,y) = 5$

Bobot garis $y - z = l(y,z) = 7$

Dari data di atas dicari bobot lintasan $W: s, x, y, z$ ($P(s,z)$), sehingga menurut Definisi 2.8, diperoleh bobot lintasan $P(s,z)$, yaitu:

$$
\begin{aligned}
P(s,z) &= l(s) + l(s,x) + l(x) + l(x,y) + l(y) + l(y,z) + l(z) \\
&= 2 + 5 + 3 + 5 + 3 + 7 + 0 = 25
\end{aligned}
$$

Jadi bobot lintasan $W: s, x, y, z$ adalah 25.

Definisi 2.9

Lintasan terpendek dari *node* i ke j adalah lintasan yang menghubungkan *node* i dan j yang mempunyai bobot lintasan minimum.

(Chartrand,G. dan O.R.Oellerman., 1993)

Sebagai gambaran, dari definisi lintasan terpendek dapat diapresiasikan pada contoh berikut:

Contoh 2.4

Perhatikan graph berbobot pada Gambar 2.3 (hal. 8), dari graph tersebut dicari lintasan yang menghubungkan *node* awal s ke *node* tujuan z yang mempunyai bobot lintasan minimal. Dalam graph itu *node* s dan z dihubungkan oleh 11 lintasan, yang masing-masing lintasan mempunyai bobot yang berbeda-beda. Dengan menghitung bobot lintasan seperti pada Contoh 2.3 maka akan diperoleh semua bobot lintasan yang menghubungkan *node* awal s dan *node* tujuan z. Data lintasan tersebut beserta bobot lintasannya disajikan dalam bentuk tabel berikut ini.

Tabel 2.3 Perhitungan bobot lintasan

No	Lintasan	Bobot lintasan (menit)	No	Lintasan	Bobot lintasan (menit)
1	$W: s, x, y, z$	25	7	$W: s, t, u, z$	25
2	$W: s, x, t, y, z$	31	8	$W: s, r, t, y, z$	30
3	$W: s, x, t, z$	27	9	$W: s, r, u, z$	26
4	$W: s, x, t, u, z$	29	10	$W: s, r, t, z$	26
5	$W: s, t, z$	23	11	$W: s, r, t, u, z$	28
6	$W: s, t, y, z$	27			

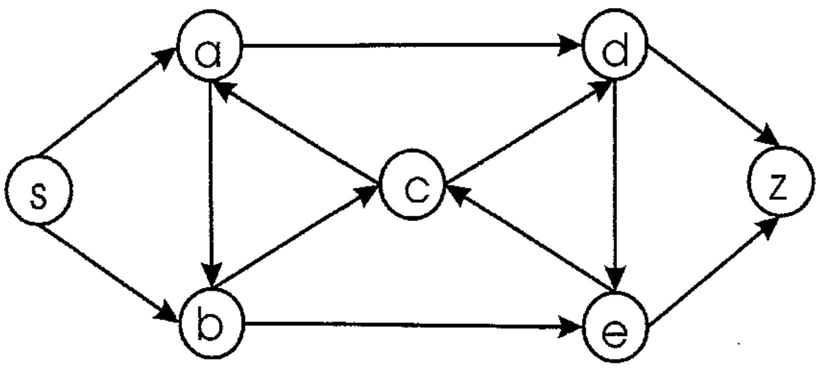
Dari lintasan-lintasan yang menghubungkan *node* awal s ke *node* tujuan z pada Tabel 2.3 akan dicari lintasan terpendek dari *node* s ke z. Dari hasil perhitungan bobot lintasan yang tertera pada Tabel 2.3 di atas diperoleh lintasan terpendek dari *node* awal s ke *node* tujuan z, yaitu lintasan $W: s, t, z$. Karena lintasan ini mempunyai bobot lintasan minimal yaitu 23, artinya waktu perjalanan minimal untuk melakukan perjalanan dari *node* s ke z adalah 23 menit.

2.2.2 *Cycle*

Definisi 2.10

Sebuah *cycle* adalah sebuah *walk* dengan n ≥ 3, dimana *node* awal sama dengan *node* akhir, $n_0 = n_n$ dengan n adalah *node* yang berbeda. (Chartrand,G. dan O.R.Oellerman., 1993)

Untuk memperjelas definisi di atas, perhatikan graph berarah berikut ini:



Gambar 2.4.
Graph berarah dengan 7 *node* dan 12 *arc.*

Dari graph di atas, yang disebut *cycle* adalah $W : b, e, c, a, b$.

Definisi 2.11

Sirkuit adalah sebuah *trail* yang mempunyai *node* awal sama dengan *node* akhir. (Chartrand,G. dan O.R.Oellerman., 1993)

Dari Definisi 2.10 dan 2.11, mengakibatkan *sebuah cycle* adalah sirkuit, tetapi sebuah sirkuit belum tentu adalah *cycle*. Sebagai gambaran, perhatikan graph berarah pada Gambar 2.4, yang disebut *cycle* adalah $W: a, d, e, c, a$, sedangkan sirkuit yang bukan *cycle* adalah $W: a, b, e, c, d, e, c, a$.

Berdasarkan bobot garis dan *node* yang menyusun sirkuit, sirkuit dapat dibedakan menjadi dua yaitu:

1. Sirkuit positif adalah suatu sirkuit dimana jumlah semua bobot garis dan *node* yang membentuk sirkuit tersebut adalah bilangan real positif.
2. Sirkuit negatif adalah suatu sirkuit dimana jumlah semua bobot garis dan *node* yang membentuk sirkuit tersebut adalah bilangan real negatif.

Dalam graph berarah yang merepresentasikan jaringan jalan dengan menggunakan lampu lalu-lintas, tidak terdapat sirkuit negatif. Hal ini berlaku karena jaringan jalan dengan menggunakan lampu lalu-lintas tidak mempunyai waktu perjalanan yang bernilai negatif. Dengan kata lain, dalam graph berarah yang merepresentasikan jaringan jalan yang menggunakan lampu lalu-lintas, tidak ada *node* atau *arc* yang berbobot negatif sehingga graph tersebut tidak mempunyai sirkuit negatif.

Definisi 2.12

Jarak *node* u ke v yang dinotasikan dengan $d(u,v)$ atau d_{uv} merupakan panjang atau bobot minimum dari seluruh *path* $u-v$ dalam graph G. (Chartrand,G. dan O.R.Oellerman., 1993)

Jika *path* $u-v$ tidak ada maka jarak dua *node* tersebut adalah tak terhingga.

Teorema mengenai jarak antara dua *node* dalam graph dapat dilihat pada Teorema 2.2 berikut ini.

Teorema 2.2

Misalkan G adalah sebuah graph, maka:

1. $d_{ij} \geq 0$, dan $d_{ij} = 0$ jika dan hanya jika $i = j$.
2. $d_{ij} = d_{ji}$ untuk semua $i, j \in N(G)$.
3. $d_{ij} \leq d_{ik} + d_{kj}$ untuk semua $i, k, j \in N(G)$.

Bukti

1. Jika d_{ij} bernilai ∞ atau sama dengan jumlah nilai semua garis yang membentuk *path* $i-j$ terdekat, maka $d_{ij} \geq 0$. Jika $d_{ij} = 0$, maka *path* $i-j$ terdekat tidak mempunyai garis, dengan demikian $i = j$, maka tidak dapat membentuk sebuah garis sehingga $d_{ij} = 0$.
2. Untuk membuktikan bahwa $d_{ij} = d_{ji}$, diasumsikan dua buah kasus, yaitu jika $d_{ij} = \infty$ dan $d_{ij} \neq \infty$. Pada kasus yang pertama, tidak ada *path* yang

menghubungkan *node* i dan j. Sedangkan pada kasus kedua, anggap *path* $i - j$ merupakan *path* terpendek, serta *path* $j - i$ diperoleh dari *path* $i - j$ dengan membalik urutan *node*. Karena *path* $j - i$ hanya merupakan pembalikan urutan *node* dari *path* $i - j$, hal ini berarti $d_{ij} = d_{ji}$.

3. Misal i, j, dan k adalah *node* dari G, dengan P adalah *path* $i - k$ terpendek dan Q adalah *path* $k - j$ terpendek pada G. Maka P diikuti Q adalah *walk* $i - j$, sebut saja W, dengan panjang $d_{ik} + d_{kj}$. Dari pengertian bahwa W memuat *path* $i - j$, maka hal ini berakibat $d_{ij} \le d_{ik} + d_{kj}$.

2.3. Graph *FIFO*

Definisi 2.13

Suatu *arc* (i, j) disebut *arc FIFO* (*First In First Out*) jika untuk setiap waktu t_h dan t_k dengan $t_h \le t_k$ berlaku $t_h + d_{ij}(t_h) \le t_k + d_{ij}(t_k)$.

(Ahuja *et al.*, 2000)

dimana t_h : waktu keberangkatan t_h.

t_k : waktu keberangkatan t_k.

$d_{ij}(t_h)$: lama perjalanan dalam *arc* (i, j) jika waktu keberangkatan t_h.

$d_{ij}(t_k)$: lama perjalanan dalam *arc* (i, j) jika waktu keberangkatan t_k.

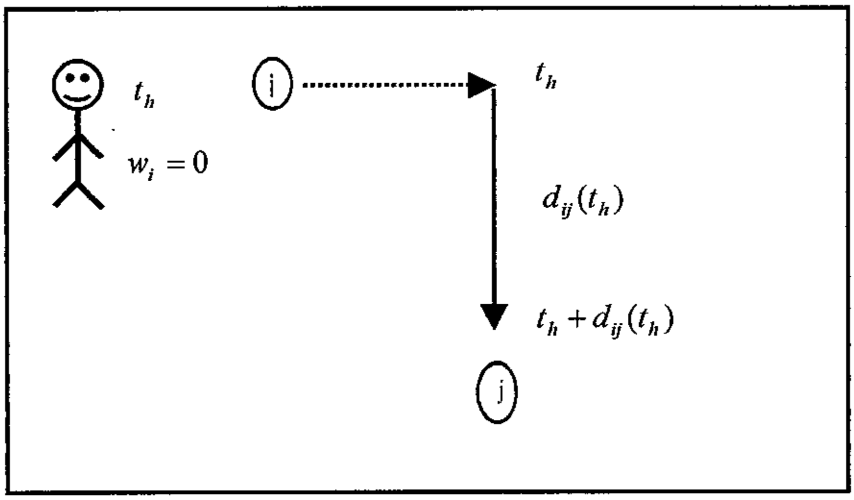
Hal ini mengakibatkan jika seseorang mulai melakukan perjalanan dalam *arc* (*i*, *j*) pada waktu t_h, maka waktu kedatangan di *node j* tidak lebih lambat dari seseorang yang perjalanannya dimulai pada waktu t_k, dengan $t_h \leq t_k$. Dengan kata lain, seseorang yang mulai melakukan perjalanan pada waktu t_k, tidak akan tiba lebih awal daripada seseorang yang perjalanannya dimulai pada waktu t_h.

Sebagai ilustrasi, berikut ini dijelaskan mengenai fenomena *arc FIFO* pada jaringan jalan. Misalkan diketahui dua buah kasus perjalanan karyawan dari *node i* ke *node j* melalui *arc* (*i*, *j*).

Kasus 1:

Karyawan pertama akan melakukan perjalanan dari *node* asal *i* ke *node* tujuan *j* melalui *arc* (*i*, *j*), dengan waktu keberangkatan dari *node i* adalah t_h.

Rencana perjalanan karyawan pertama diperlihatkan oleh gambar di bawah ini:



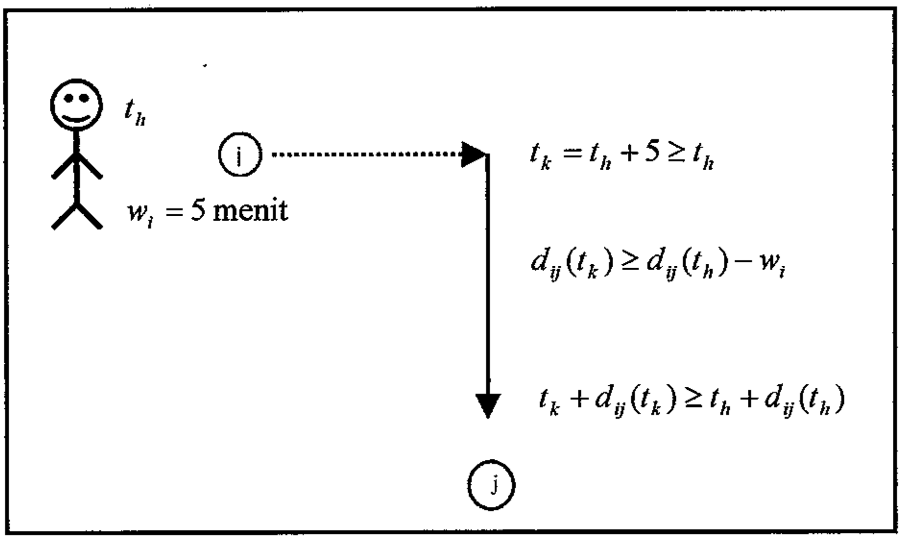
Gambar 2.5
Diagram perjalanan karyawan I

Seperti yang diilustrasikan oleh gambar tersebut, karyawan pertama tidak memutuskan untuk menunggu pada *node i*, sehingga waktu tunggu karyawan pertama pada *node i* yang dinotasikan dengan w_i adalah nol, ($w_i = 0$). Hal ini menyebabkan perjalanan karyawan pertama dimulai ketika waktu menunjukkan t_h. Jika diketahui $d_{ij}(t_h)$ adalah waktu yang diperlukan untuk menempuh *arc* *(i, j)*, maka kedatangan karyawan yang pertama pada *node j* adalah ketika waktu menunjukkan $t_h + d_{ij}(t_h)$.

Kasus 2:

Karyawan kedua akan melakukan perjalanan yang sama dengan karyawan yang pertama, tetapi dengan waktu keberangkatan yang berbeda, yaitu t_k.

Rencana perjalanan karyawan yang kedua ditunjukkan oleh gambar berikut ini.



Gambar 2.6
Diagram perjalanan karyawan II

Dari diagram perjalanan karyawan kedua pada gambar di atas dapat dijelaskan bahwa sebelum memulai perjalanan dalam *arc* *(i, j)*, karyawan yang kedua mengambil keputusan untuk menunggu dalam *node i* selama 5 menit. Hal ini mengakibatkan waktu tunggu karyawan kedua pada *node i* adalah selama lima menit, atau berlaku $w_i = 5\,\text{menit}$. Tujuan karyawan kedua menunggu pada *node i* adalah supaya waktu perjalanan dalam *arc* (*i*, *j*) menjadi lebih kecil. Keputusan tersebut menyebabkan perjalanan yang dilakukan oleh karyawan kedua dimulai pada saat $t_h + 5\,\text{menit} = t_k$. Karena dalam *arc* ini berlaku syarat *FIFO*, maka waktu perjalanan karyawan kedua dalam *arc* (*i*, *j*), yang dinotasikan dengan $d_{ij}(t_k)$, bisa lebih sedikit dari $d_{ij}(t_h)$, tetapi tidak berkurang lebih dari 5 menit.

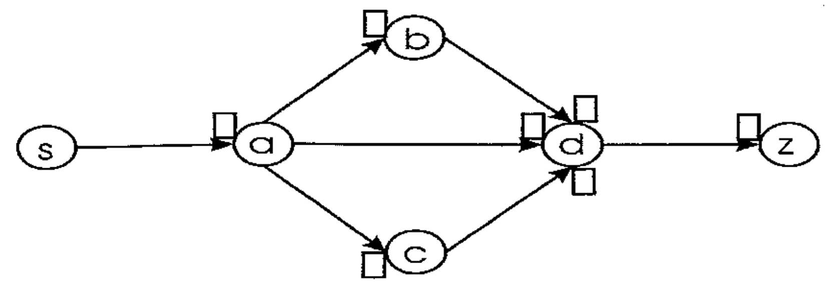
Hal ini menyebabkan waktu kedatangan karyawan pertama pada *node j* tidak akan lebih lambat dari karyawan yang kedua.

Dari fenomena tersebut dapat diambil suatu kesimpulan, pada *arc* yang memenuhi syarat *FIFO*, keputusan untuk menunggu terlebih dahulu pada *node* awal sebelum melakukan perjalanan tidak memberikan keuntungan. Karena waktu kedatangan di *node* tujuan tidak akan lebih cepat dari keadaan sebelumnya, dimana perjalanan dilakukan tanpa menunggu pada *node* awal terlebih dahulu.

Definisi 2.14

Suatu graph disebut graph *FIFO* jika semua *arc* dalam graph tersebut adalah *arc FIFO*. (Ahuja *et al.*, 2000)

Untuk lebih jelasnya, perhatikan jaringan jalan yang menggunakan lampu lalu-lintas, yang direpresentasikan dalam graph berarah berikut ini.



Keterangan gambar:

: lingkaran menggambarkan *node*

: garis berarah menggambarkan jalan

: kotak mengambarkan lampu lalu-lintas

Gambar 2.7
Jaringan jalan yang menggunakan lampu lalu-lintas

Jaringan jalan yang direpresentasikan oleh graph berarah pada Gambar 2.7 merupakan jaringan dengan graph *FIFO*. Teorema berikut ini membuktikan bahwa jaringan tersebut merupakan jaringan dengan graph *FIFO*.

Teorema 2.3

Misalkan G adalah graph yang merepresentasikan jaringan jalan yang menggunakan lampu lalu-lintas, dengan waktu yang diperlukan untuk melakukan perjalanan pada setiap *arc* dari graph G adalah konstan, maka G adalah graph *FIFO*.

Bukti

Dari Gambar 2.7, yang menggambarkan jaringan jalan yang menggunakan lampu lalu-lintas, diketahui:

t_h : waktu keberangkatan t_h.

t_k : waktu keberangkatan t_k.

$d_{sa}(t_h)$: lama perjalanan dalam *arc* (*s*, *a*) jika waktu keberangkatan t_h.

$d_{sa}(t_k)$: lama perjalanan dalam *arc* (*s*, *a*) jika waktu keberangkatan t_k.

$w(s,a,t_h+d_{sa}(t_h))$: waktu tunggu pada *node a*, jika waktu keberangkatan dari *node s* adalah t_h.

$w(s,a,t_k+d_{sa}(t_k))$: waktu tunggu pada *node a*, jika waktu keberangkatan dari *node s* adalah t_k.

$$g(t_h) = t_h + d_{sa}(t_h) + w(s,a,t_h + d_{sa}(t_h))$$

$$g(t_k) = t_k + d_{sa}(t_k) + w(s,a,t_k + d_{sa}(t_k))$$

Diberikan asumsi :

a. $t_h \le t_k$

b. Waktu yang diperlukan untuk melakukan perjalanan pada setiap *arc* dari graph *G* adalah konstan.

c. Apabila waktu kedatangan kendaraan di *node a* adalah fase hijau, maka kendaraan langsung melanjutkan perjalanan tanpa menunggu di *node a*.

Jika waktu kedatangan di *node* a adalah fase merah maka kendaraan harus menunggu sampai mulainya fase berikutnya dan kemudian langsung melanjutkan perjalanan.

Dari asumsi-asumsi tersebut mengakibatkan pada *arc* $e=(s,a)\in \mathrm{A(G)}$, berlaku kondisi sebagai berikut:

1. $t_h \le t_k$
2. $d_{sa}(t_h)=d_{sa}(t_k)=d_{sa}(t)$
3. $\{w(s,a,t_h+d_{sa}(t_h))-w(s,a,t_k+d_{sa}(t_k))\}\le t_k-t_h$

Akan dibuktikan bahwa *arc* $e=(s,a)\in \mathrm{A(G)}$, merupakan *arc FIFO.* Ditunjukkan untuk setiap $t_h \le t_k$ berlaku $g(t_h)-g(t_k)\le 0$.

$$g(t_h)-g(t_k)=$$
$$=\{t_h+d_{sa}(t_h)+w(s,a,t_h+d_{sa}(t_h))\}-\{t_k+d_{sa}(t_k)+w(s,a,t_k+d_{sa}(t_k))\}$$
$$=\{\underbrace{[t_h-t_k]}_{\le 0}+\underbrace{[d_{sa}(t_h)-d_{sa}(t_k)]}_{0}+\underbrace{[w(s,a,t_h+d_{sa}(t_h))-w(s,a,t_k+d_{sa}(t_k))]}_{\le t_k-t_h}\}$$
$$g(t_h)-g(t_k)\le 0$$

Terbukti bahwa *arc* $e=(s,a)\in \mathrm{A(G)}$ memenuhi syarat *FIFO*, sehingga *arc* $e=(s,a)\in \mathrm{A(G)}$ merupakan *arc FIFO.*

Untuk *arc* yang lain dalam graph *G*, dengan prosedur yang sama dapat diperoleh kondisi dimana semua *arc* dalam graph *G* memenuhi syarat *FIFO.* Dengan demikian jaringan jalan yang mengunakan lampu lalu-lintas, merupakan jaringan dengan graph *FIFO.*

2.4. Jaringan transportasi pada graph berarah

Definisi 2.15

Suatu graph berbobot dinamakan jaringan transportasi (*transport network*) jika sejumlah syarat-syarat berikut ini dipenuhi:

1. Graph tersebut terhubung dan tidak mempunyai *loop*.
2. Terdapat satu dan hanya satu *node* di dalam graph tersebut yang tidak mempunyai *arc* masuk.
3. Terdapat satu dan hanya satu *node* di dalam graph tersebut yang tidak mempunyai *arc* keluar.
4. Pembobot setiap *arc* berupa sebuah bilangan tidak negatif.

(Chartrand,G. dan O.R.Oellerman., 1993)

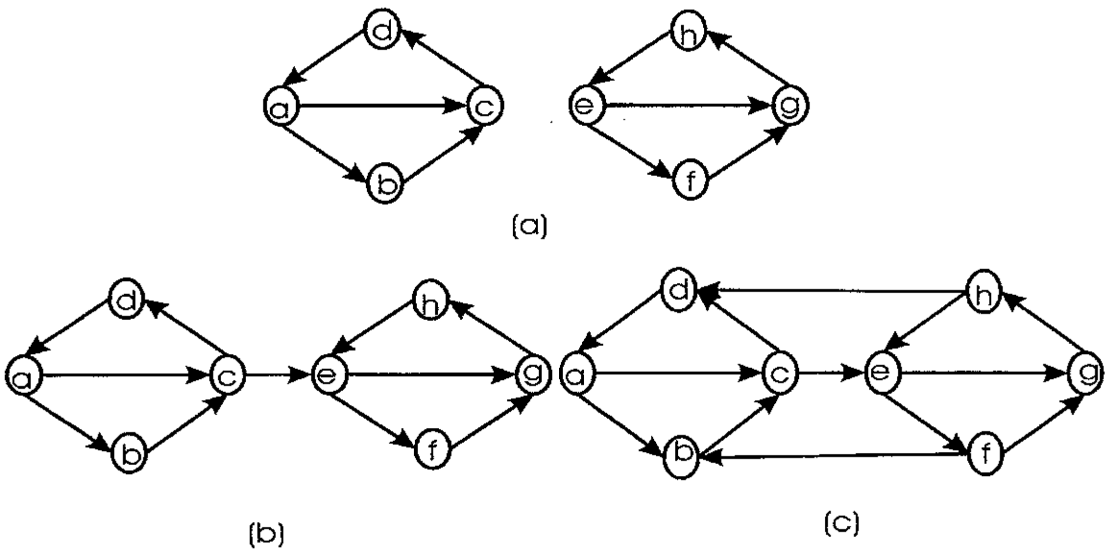
Di dalam graph yang menggambarkan jaringan transportasi, *node* yang tidak mempunyai *arc* masuk dinamakan sumber (*source*) dan dilambangkan dengan *s*, *node* yang tidak mempunyai *arc* keluar disebut tujuan (*slink*) dan dinotasikan dengan *j*.

Definisi 2.16

Dua *node* *i* dan *j* dikatakan terhubung jika graphnya paling sedikit memuat sebuah *path* yang menghubungkan *node* *i* dan *j*.

(Chartrand,G. dan O.R.Oellerman., 1993)

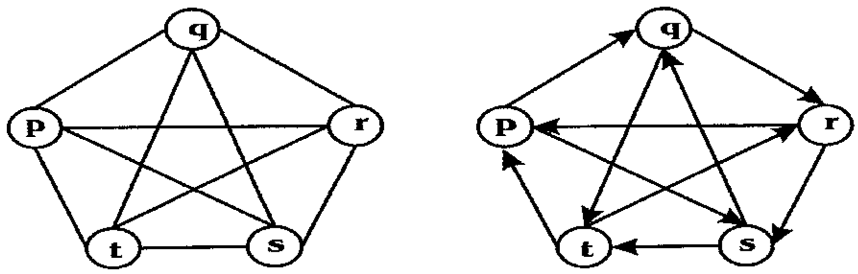
Sebuah graph dikatakan terhubung jika setiap pasangan *node* terhubung dan dikatakan tak terhubung jika ada *node* yang tak terhubung. Graph berarah disebut terhubung kuat jika ada lintasan di graph *G* dari setiap *node* ke setiap *node* lainnya.



Gambar 2.8
(a). Graph tidak terhubung
(b). Graph terhubung
(c). Graph terhubung kuat

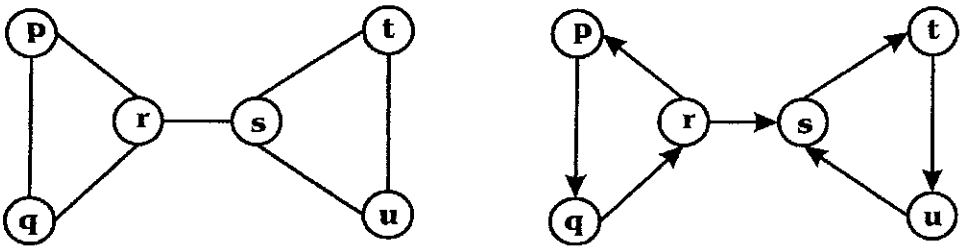
Sebagai ilustrasi, pengemudi yang akan berkeliling suatu kota yang mempunyai sistem jalan raya satu arah. Misalkan jaringan jalan yang terdapat pada kota tersebut, direpresentasikan kedalam graph berarah seperti pada Gambar 2.8. Jika graph itu adalah terhubung kuat (perhatikan Gambar 2.8c), berarti pengemudi dapat menjangkau setiap bagian kota dengan mudah, dengan hanya mengikuti arah jalan-jalan tersebut. Tetapi apabila graphnya hanya terhubung (lihat Gambar 2.8b), berarti tidak mudah untuk bepergian dari suatu bagian kota ke bagian kota lainnya.

Garis-garis pada graph $G(N,E)$ dalam gambar 2.9a, dapat diberi arah sedemikian sehingga diperoleh graph berarah yang terhubung kuat, seperti yang ditunjukkan oleh Gambar 2.9b. Sebaliknya tidak mungkin untuk memberi arah garis-garis pada graph dalam Gambar 2.10a, sedemikian sehingga diperoleh graph berarah yang terhubung kuat. Dengan kata lain graph berarah yang diperoleh merupakan graph yang tidak terhubung kuat, yang ditunjukkan oleh Gambar 2.10b. Hal ini terjadi karena garis yang merupakan 'jembatan' yaitu garis r - s harus memiliki arah tertentu.



(a) (b)

Gambar 2.9
a. Graph dengan 5 *node* dan 10 *edge*
b. Graph berarah yang terhubung kuat



(a) (b)

Gambar 2.10
a. Graph dengan 6 *node* dan 7 *edge*
b. Graph berarah yang tidak terhubung kuat

2.5. *Modular aritmatik* n

Definisi 2.17

Misalkan n bilangan bulat positif tertentu. Untuk sembarang bilangan bulat a dan b, (a + b) mod n (dibaca "a ditambah b *modulo* n") adalah sisa pambagian (a + b) oleh n; (a x b) mod n (dibaca "a kali b *modulo* n") adalah sisa pambagian (a x b) oleh n. (Whitelaw, TA., 1995)

Untuk lebih jelasnya perhatikan contoh berikut ini.

Contoh 2.6

(7 + 4) mod 3 = 11 mod 3 = 2, artinya sisa pembagian (7 + 4) oleh 3 adalah 2.

(7 - 4) mod 3 = 3 mod 3 = 0, artinya sisa pembagian (7 - 4) oleh 3 adalah 0.

(7 x 4) mod 3 = 28 mod 3 = 1, artinya sisa pambagian (7 x 4) oleh 3 adalah 1.

Definisi tentang *modular aritmatik* n di atas, diperlukan untuk memperoleh waktu tunggu di setiap persimpangan jalan, pada permasalahan rute terpendek dalam jaringan jalan yang menggunakan lampu lalu-lintas.