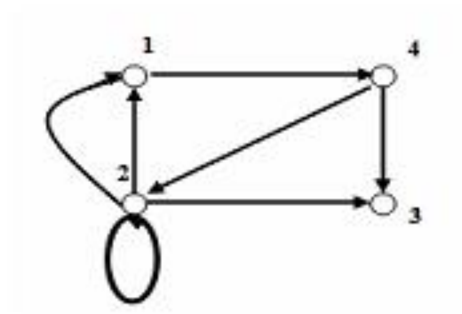
Graf Berarah (Digraf)

Di dalam situasi yang dinamis, seperti pada komputer digital ataupun pada sistem aliran (flow system), konsep graf berarah lebih sering digunakan dibandingkan dengan konsep graf tak berarah.



Apabila ruas suatu graf berarah mempunyai suatu bobot, graf berarah tersebut dinamakan suatu jaringan atau *network*.

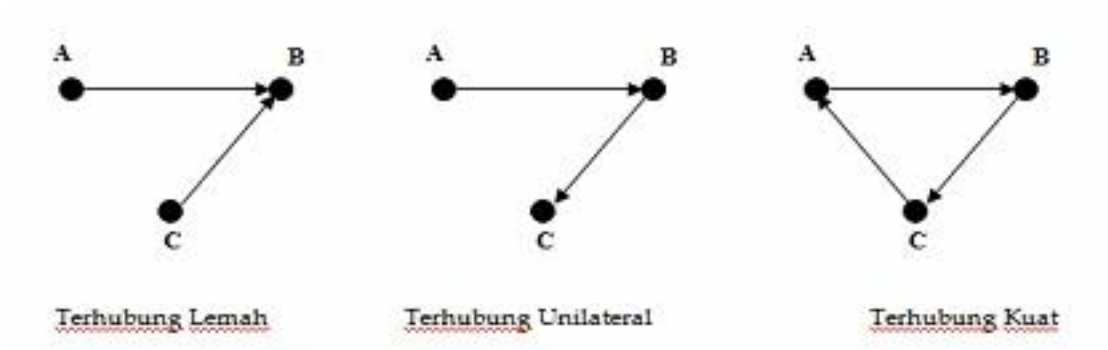
Beberapa Pengertian dalam graf berarah :

- Derajat ke luar (out degree) suatu simpul adalah banyaknya ruas yang mulai / keluar dari simpul tersebut.
- Derajat ke dalam (in degree) suatu simpul adalah banyaknya ruas yang berakhir / masuk ke simpul tersebut.
- Simpul berderajat ke dalam = 0 disebut sumber (source), sedangkan simpul berderajat ke luar = 0 disebut muara (sink).
- Pengertian Walk, Trail, Path (Jalur) dan Sirkuit (Cycle) berlaku pula pada graf berarah, dimana harus sesuai dengan arah ruas. Kalau tidak sesuai dengan arah ruas-nya, maka disebut sebagai semi walk, semi path atau semi trail.

Pada graf berarah terdapat 3 pengertian keterhubungan, yakni :

- Terhubung lemah, jika terdapat suatu semi path antara setiap 2 simpul dari D.
- Terhubung unilateral, jika antara setiap 2 simpul u dan v dari D, terdapat jalur dari u ke v atau dari v ke u.
- Terhubung kuat, jika antara setiap 2 simpul u dan v dari D, terdapat jalur dari u ke v dan dari v ke u.

Contoh :



RELASI DAN MATRIKS

Pandang D(V,A) suatu graf berarah tanpa ruas sejajar, maka A adalah himpunan bagian dari V x V (produk Cartesis himpunan), jadi merupakan Relasi pada V. Sebaliknya bila R adalah Relasi pada suatu himpunan V, maka D(V,R) merupakan graf berarah tanpa ruas sejajar. Jadi konsep Relasi dan konsep graf berarah tanpa ruas sejajar adalah satu dan sama.

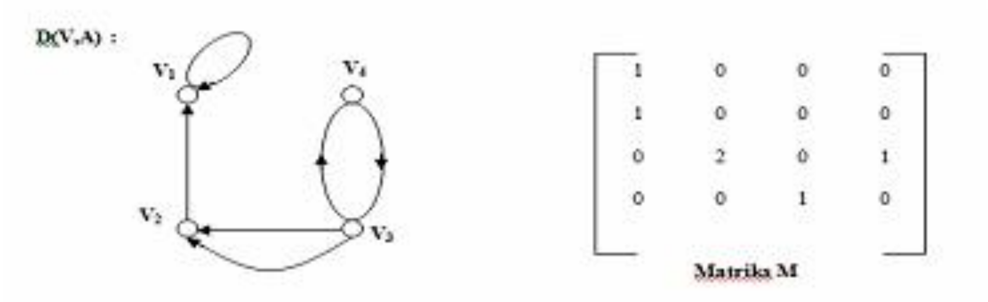
Misalkan D(V,A) suatu graf berarah dengan simpul v_1, v_2, … , v_m. Matriks M berukuran (mxm) merupakan matriks (matriks adjacency) dari D, dengan mendefinisikan sebagai berikut :

M = (M_{ij}), dengan m_{ij} banyaknya ruas yang mulai di v_i dan berakhir di v_j

Bila D tidak mengandung ruas berganda, maka elemen M adalah 0 dan 1. Kalau Graf berarah mengandung ruas berganda, elemen M merupakan bilangan bulat non negatif.

Jadi suatu matriks berukuran (mxm) yang elemennya bilangan bulat non negatif menyatakan secara tunggal suatu graf berarah dengan m simpul.

Contoh :



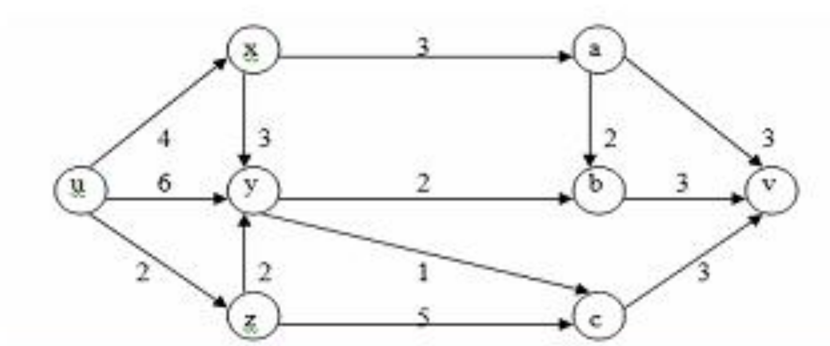
Teorema :

M adalah Matriks dari sutau graf berarah D, maka elemen baris ke i kolom ke j dari Matriks M $_n$ menyatakan banyaknya walk dengan panjang n dari simpul v_i ke simpul v_j.

ALGORITMA JALUR TERPENDEK

Pandang D suatu Graf berarah yang hingga dengan tiap-tiap ruas mempunyai bobot. Jadi D merupakan suatu Network. Kita hendak menentukan Jalur Terpendek antara simpul u dan v. Misalkan D tidak mengandung sirkuit.

Sebagai contoh, gambar berikut merupakan suatu Network. Kita hendak menghitung Jalur terpendek dari simpul u ke v.



Simpul u disebut **Sumber (Source)**.

Simpul v disebut **Muara (Sink)**.

Untuk menentukan Jalur Terpendek tersebut, cara berikut dapat digunakan :

- Buat tabel jarak :

u	x	y	z	a	b	c	v
ux = 4 uy = 6 uz = 2	xy = 3 xa = 3	yb = 2 yc = 1	zy = 2 zc = 5	ab = 2 av = 3	bv = 3	cv = 3	

- Kita mulai dengan simpul u sebagai simpul awal. Beri harga = 0. Ambil simpul dengan jarak terdekat dari u (dalam hal ini z = 2), kemudian lingkari uz. Semua ruas lain yang berakhir di z kita hapus (dalam hal ini tidak ada ruas lain yang berakhir di z. Beri nilai = 2 di belakang z. Simpul yang telah diberi harga ditandai dengan *.

u*(0)	x	y	z*(2)	a	b	c	v
ux = 4 uy = 6 uz = 2	xy = 3 xa = 3	yb = 2 yc = 1	zy = 2(4) zc = 5(7)	ab = 2 av = 3	bv = 3	cv = 3	

- Dari simpul u dan z (yang telah ditandai *), dicari simpul lain yang jarknya terdekat dihitung dari u. Jadi harus diperhitungkan nilai yang tertulis di simpul (0 untuk u dan 2 untuk z). Disini ux = 4 dan uzy = 2 + 2 = 4 merupakan nilai minimal. Boleh dipilih salah satu, misalnya uzy. Beri nilai = 4 pada y. Lingkari zy dan hapus ruas yang lain yang menuju y,

yaitu uy dan xy.

u*(0)	x	y	z*(2)	a	b	c	v
ux = 4 ~~uy = 6~~ uz = 2	~~xy = 3~~ xa = 3	yb = 2 yc = 1	zy = 2(4) zc = 5(7)	ab = 2 av = 3	bv = 3	cv = 3	

- Demikian proses dilanjutkan berturut-turut :

u*(0)	x	y*(4)	z*(2)	a	b	c	v
ux = 4 ~~uy = 6~~ uz = 2	~~xy = 3~~ xa = 3	yb = 2(6) yc = 1(5)	zy = 2(4) zc = 5(7)	ab = 2 av = 3	bv = 3	cv = 3	

u*(0)	x*(4)	y*(4)	z*(2)	a	b	c	v
ux = 4 ~~uy = 6~~ uz = 2	~~xy = 3~~ xa = 3(7)	yb = 2(6) yc = 1(5)	zy = 2(4) zc = 5(7)	ab = 2 av = 3	bv = 3	cv = 3	

u*(0)	x*(4)	y*(4)	z*(2)	a	b	c*(5)	v
ux = 4 ~~uy = 6~~ uz = 2	~~xy = 3~~ xa = 3(7)	yb = 2(6) yc = 1(5)	zy = 2(4) ~~zc = 5(7)~~	ab = 2 av = 3	bv = 3	cv = 3(8)	

u*(0)	x*(4)	y*(4)	z*(2)	a	b*(6)	c*(5)	v
ux = 4 ~~uy = 6~~ uz = 2	~~xy = 3~~ xa = 3(7)	yb = 2(6) yc = 1(5)	zy = 2(4) ~~zc = 5(7)~~	~~ab = 2~~ av = 3	bv = 3(9)	cv = 3(8)	

u*(0)	x*(4)	y*(4)	z*(2)	a*(7)	b*(6)	c*(5)	v
ux = 4 ~~uy = 6~~ uz = 2	~~xy = 3~~ xa = 3(7)	yb = 2(6) yc = 1(5)	zy = 2(4) zc = 5(7)	~~ab = 2~~ av = 3(10)	bv = 3(9)	cv = 3(8)	

u*(0)	x*(4)	y*(4)	z*(2)	a*(7)	b*(6)	c*(5)	v*(8)
ux = 4 uy = 6 uz = 2	xy = 3 xa = 3(7)	yb = 2(6) yc = 1(5)	zy = 2(4) ~~zc = 5(7)~~	~~ab = 2~~ ~~av = 3(10)~~	~~bv = 3(9)~~	cv = 3(8)	

- Diperoleh jalur minimal dari simpul u ke simpul v yang **panjangnya = 8** dengan urutan

$$\mathbf{v \leftarrow c \leftarrow y \leftarrow z \leftarrow u}$$

Algoritma diatas dapat pula dikenakan untuk Graf tidak berarah.

PROBLEMA ALIRAN MAKSIMAL

Tujuan dari Problema Aliran Maksimal adalah :

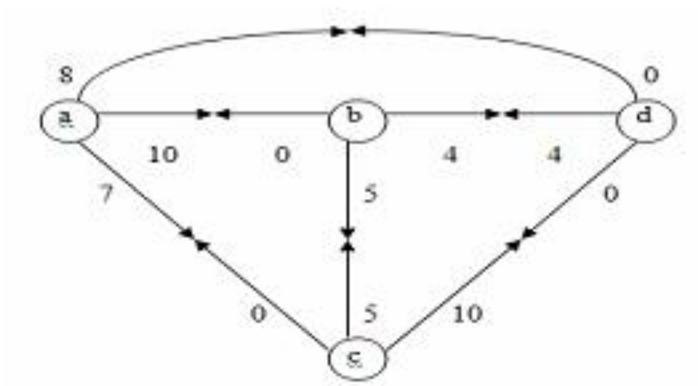
Mengatur jadwal pengiriman barang agar jumlah barang yang dikirimkan dari suatu simpul ke simpul lain (yang tertentu) adalah maksimum.

Simpul yang mengirimkan (simpul awal) disebut **Sumber (Source)**.

Simpul yang menerima kiriman (simpul akhir) disebut **Muara (Sink)**.

Antara Sumber dan Muara terdapat pula simpul lain yang disebut **Simpul Perantara**.

Dalam hal ini ditetapkan bahwa simpul perantara tidak dapat menyimpan barang.



Perharikan Graf diatas. Simpul a adalah Sumber. Simpul d adalah Muara. Sedangkan simpul b dan c adalah Simpul Perantara. Angka pada masing-masing ruas menyatakan kapasitas ruas tersebut.

Jadi, misalkan dari a dapat dikirimkan 10 buah/unit barang ke b, sedangkan dari b tidak dapat dikirimkan barang ke a.

Untuk menyelesaikan problema aliran maksimal diatas, dapat kita gunakan suatu algoritma.

Algoritma Problema Aliran Maksimal adalah sebagai berikut :

1) Cari suatu jalur dari Sumber ke Muara yang dapat membawa aliran barang yang positif.

   Kalau tak ada, langsung ke langkah (4). Tentukan aliran maksimal jalur tersebut.

   Contoh : Pada problema diatas dapat diambil jalur **ad**.

   Aliran maksimum jalur tersebut adalah 8.

2) Pada graf berikutnya kapasitas ruas pada jalur kita kurangi dengan aliran maksimum, dan kapasitas ruas yang berlawanan bertambah dengan aliran maksimum tersebut.

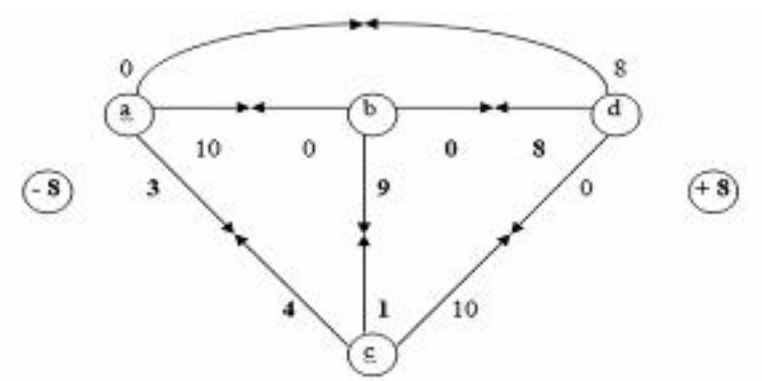
   Contoh : Pada contoh kita, kapasitas ruas **ad** menjadi 8 **-** 8 = 0

   Dan kapasitas ruas **da** menjadi 0 + 8 = 8

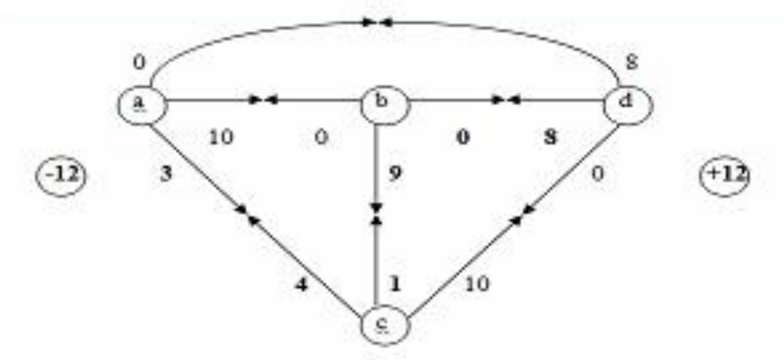
3) Kembali ke langkah (1).

4) Aliran Maksimum adalah jumlah semua barang yang diterima oleh Muara.

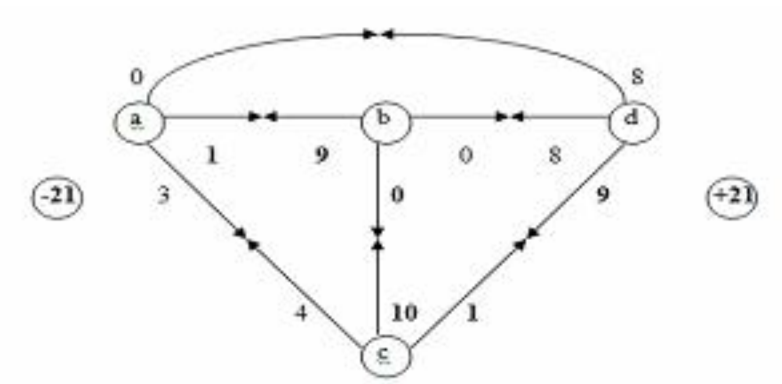
Berikut ini adalah penyelesaian problema di atas :



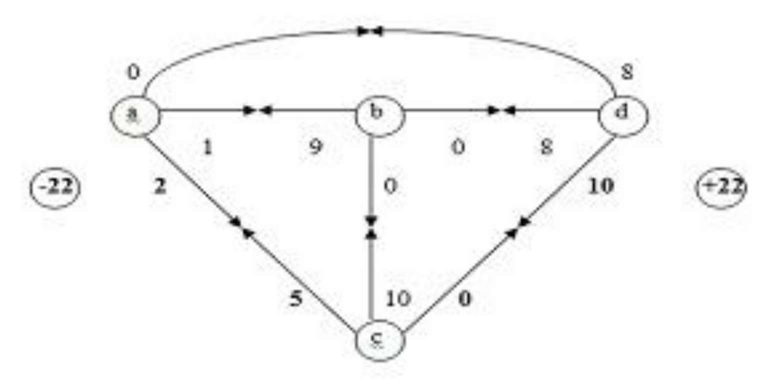
Jalur **ad**, aliran maksimal = 8



Jalur **acbd**, aliran maksimal = 4



Jalur **abcd**, aliran maksimal = 9



Jalur **acd**, aliran maksimal = 1

Tak ada lagi Jalur dari Sumber ke Muara yang dapat membawa aliran positif. Jadi diperoleh **aliran maksimal dari jaringan adalah 22**.

MESIN STATE HINGGA

Mesin State Hingga merupakan suatu struktur abstrak yang didefinisikan terdiri atas :

(1) Himpunan hingga A, berisi simbol input

(2) Himpunan hingga S, berisi internal state

(3) Himpunan hingga Z, berisi simbol output

(4) Sebuah fungsi f : S x A $\rightarrow$ S, disebut fungsi next-state

(5) Seubuah fungsi g : S x A $\rightarrow$ Z disebut fungsi output

$\Rightarrow$ M (A, S, Z, f, g)

$\Rightarrow$ M (A, S, Z, q0, f, g)

INPUT	:	Untai
OUTPUT	:	Untai

Contoh : M (A, S, Z, f, g) dengan :

(1) A = (a,b)

(2) S = (q0, q1, q2)

(3) Z = (x, y, z)

(4) f : S x A $\rightarrow$ S, yang didefinisikan sebagai :

f (qo, a) = q1	f (q0, b) = q2
f (q1, a) = q2	f (q1, b) = q1
f (q2, a) = qo	f (q2, b) = q1

(5) g : S x A $\rightarrow$ Z, yang didefinisikan sebagai :

g (q0, a) = x	g (q0, b) = y
g (q1, a) = x	g (q1, b) = z
g (q2, a) = z	g (q2, b) = y

AUTOMATA HINGGA

Automata Hingga merupakan suatu struktur abstrak yang didefinisikan terdiri atas :

(1) Himpunan hingga A, berisi simbul input

(2) Himpunan hingga S, berisi internal state

(3) Himpunan T (dimana T $\subset$ S), elemennya disebut state penerima

(4) State awal (biasanya q0), anggota S

(5) Fungsi next-state f : S x A $\rightarrow$ S

$\Rightarrow$ M (A, S, T, qo, f)

INPUT : Untai
OUTPUT : Diterima atau ditolak

Contoh : M (A, S, T, qo, f) dengan :

(1) A = { a, b }

(2) S = { q0, q1, q2 }

(3) T = { qo, q1 }

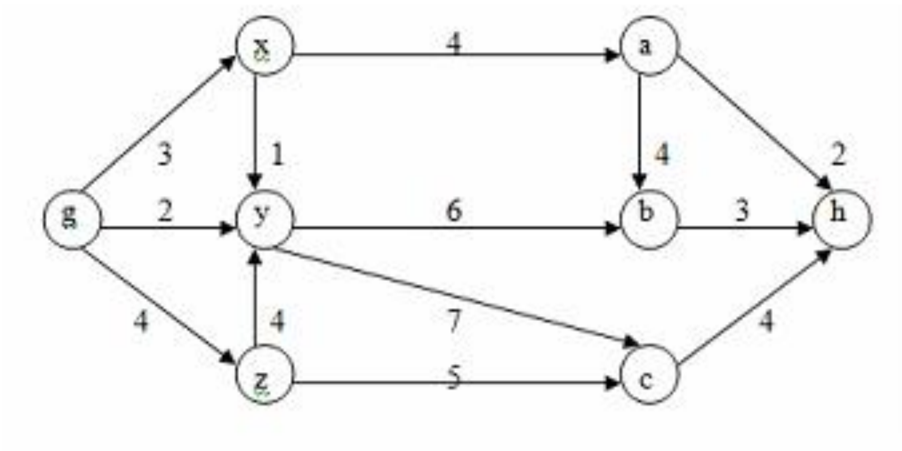
(4) State awal = q0

(5) Fungsi next-state f : S x A $\rightarrow$ S, yang didefinisikan sebagai tabel berikut :

f	**a**	**b**
q0	q0	q1
q1	q0	q2
q2	q2	q2

LATIHAN

1. Tentukan jalur terpendek dari G ke H!



2. Selesaikanlah problema aliran maksimal dari graph berikut. Simpul x merupakan sumber dan y merupakan muara!