GRAF

(*Slide Acknowledgment*: Gatot Wahyudi, Adila A. Krisnadhi, Kurniawati Azizah)

Matematika Diskret 2

Fakultas Ilmu Komputer Universitas Indonesia

Agenda

- Isomorfisme
- Bipartite dan Matching
- Lintasan Terpendek

Isomorfisme

Isomorfisme

- Definisi

> Dua graf sederhana $G_1 = (V_1, E_1)$ dan $G_2 = (V_2, E_2)$ dikatakan isomorfik jika terdapat fungsi bijektif f dari V_1 ke V_2 sehingga dua vertex v_1 dan v_2 bertetangga di G_1 jika dan hanya jika vertex $f(v_1)$ dan $f(v_2)$ bertetangga di G_2.
>
> Jika tidak ada isomorfisme antara G_1 dan G_2, maka G_1 dan G_2 dikatakan nonisomorfik.

- Jika $G_1 = (V_1, E_1)$ dan $G_2 = (V_2, E_2)$ isomorfik, maka $|V_1| = |V_2|$, $|E_1| = |E_2|$ dan untuk setiap $n \in \mathbb{N}$, $|\{u \in V_1 | \deg(u) = n\}| = |\{v \in V_2 | \deg(v) = n\}|$.
  (Konversnya belum tentu berlaku).
- Untuk mengecek apakah suatu fungsi dari V_1 ke V_2 adalah isomorfisme, kita dapat menggunakan matriks ketetanggaan (matriks ketetanggaan kedua graf harus serupa).

Menentukan Isomorfisme

- Untuk mengetahui dua buah graf sederhana bersifat isomorfik, perlu dicari fungsi bijektif yang memetakan V_1 ke V_2 sedemikian hingga ketetanggaan di kedua graf tetap terjaga.
- Jika kedua graf memiliki n jumlah simpul, maka terdapat $n!$ kemungkinan fungsi bijektif yang dapat dibangun.
- Untuk menentukan dua graf non-isomorfik, kita dapat memeriksa adanya pelanggaran terhadap invarian graf.
- **Invarian graf** adalah properti pada graf yang menjaga sifat isomorfisme graf tersebut.
- Contoh invarian graf:
    - jumlah simpul
    - jumlah sisi
    - jumlah loop
    - jumlah simpul dengan derajat tertentu

Contoh penentuan isomorfisme (1)

Apakah dua graf $G = (V, E)$ dan $H = (W, F)$ isomorf apabila:

- $V = \{ v_1, v_2, v_3, v_4, v_5, \}$
- $E = \{ (v_1, v_2), (v_2, v_3), (v_3, v_4), (v_4, v_1), (v_1, v_5) \}$
- $W = \{ w_1, w_2, w_3, w_4, w_5, \}$
- $F = \{ (w_1, w_3), (w_1, w_4), (w_2, w_5), (w_4, w_2), (w_5, w_1) \}$

- Jawab

  Untuk menyelidiki apakah dua graf isomorf, harus dicari <u>fungsi satu-satu</u> dari V ke W sehinga keterikatan dua vertex dapat dipertahankan

Contoh penentuan isomorfisme (2)

- Langkah 1, periksa $|V| = |W|$ dan $|E| = |F|$
  - Jika TIDAK, maka kedua graf jelas tidak isomorfik
  - Jika YA, maka lanjutkan ke pemeriksaan lebih lanjut

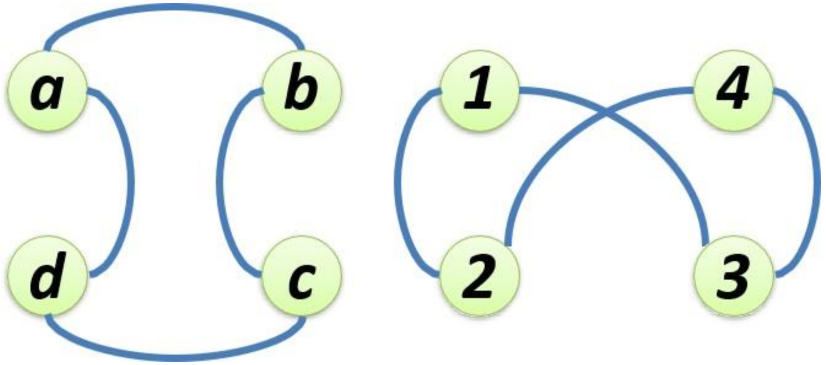
- Langkah 2, tentukan derajat masing-masing vertex di V dan W
  - $\deg(v_1) = 3$, $\deg(v_2) = 2$, $\deg(v_3) = 2$, $\deg(v_4) = 2$, $\deg(v_5) = 1$
  - $\deg(w_1) = 3$, $\deg(w_2) = 2$, $\deg(w_3) = 1$, $\deg(w_4) = 2$, $\deg(w_5) = 2$

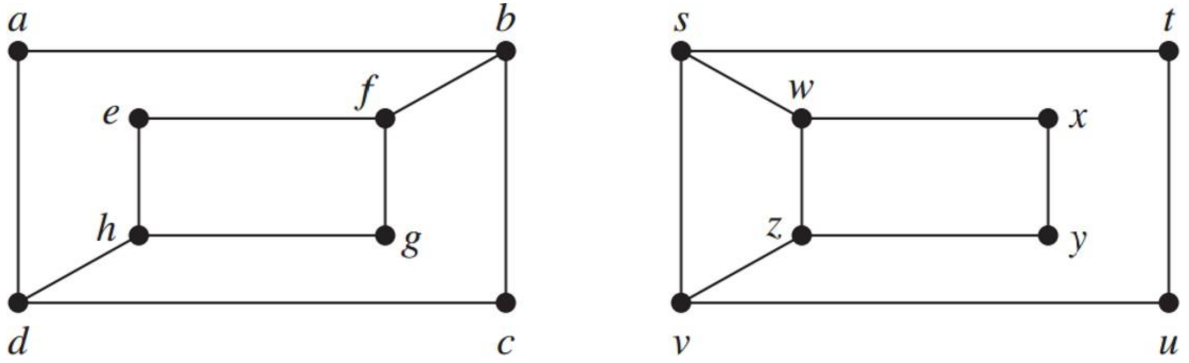
Contoh penentuan isomorfisme (3)

- Langkah 3
  - Dengan memanfaatkan informasi derajat masing-masing vertex, kita coba mencari fungsi bijektif yang dimaksud. Misalnya, kita coba fungsi bijektif berikut:
    - $f(v_1) = w_1,\ f(v_2) = w_4,\ f(v_3) = w_2,\ f(v_4) = w_5,\ f(v_5) = w_3$
  - Selanjutnya, kita lihat keterpeliharaan keterikatan antar dua vertex sebagai berikut:
    - $(v_1, v_2) \in E$ maka $(f(v_1), f(v_2)) = (w_1, w_4) \in F$,
    - $(v_2, v_3) \in E$ maka $(f(v_2), f(v_3)) = (w_4, w_2) \in F$,
    - $(v_3, v_4) \in E$ maka $(f(v_3), f(v_4)) = (w_2, w_5) \in F$,
    - $(v_4, v_1) \in E$ maka $(f(v_4), f(v_1)) = (w_5, w_1) \in F$,
    - $(v_1, v_5) \in E$ maka $(f(v_1), f(v_5)) = (w_1, w_3) \in F$,
  - Ternyata semua terpenuhi, maka G dan H isomorf.

Latihan

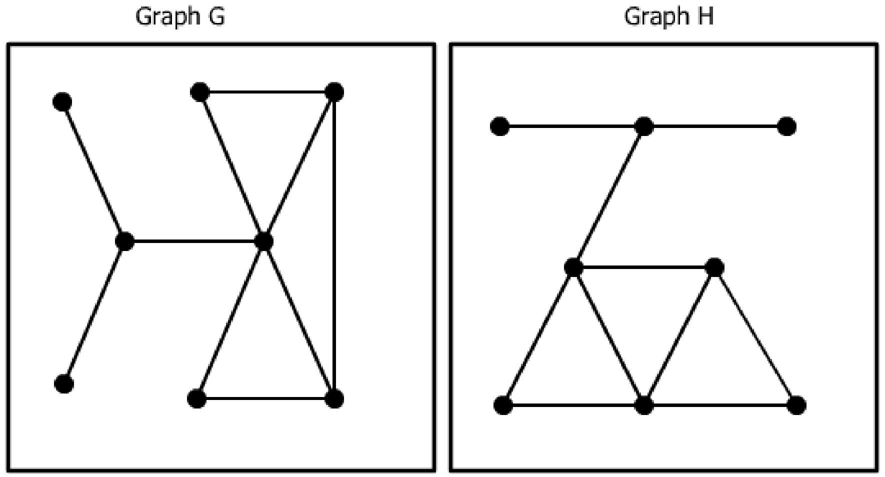
- Tentukan apakah pasangan-pasangan graf berikut bersifat isomorfik.





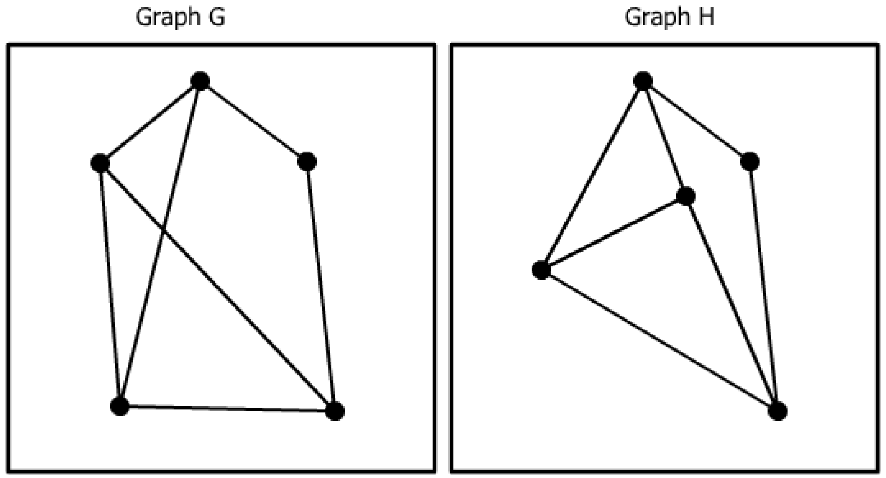
Latihan

- Tentukan apakah pasangan graf berikut bersifat isomorfik.



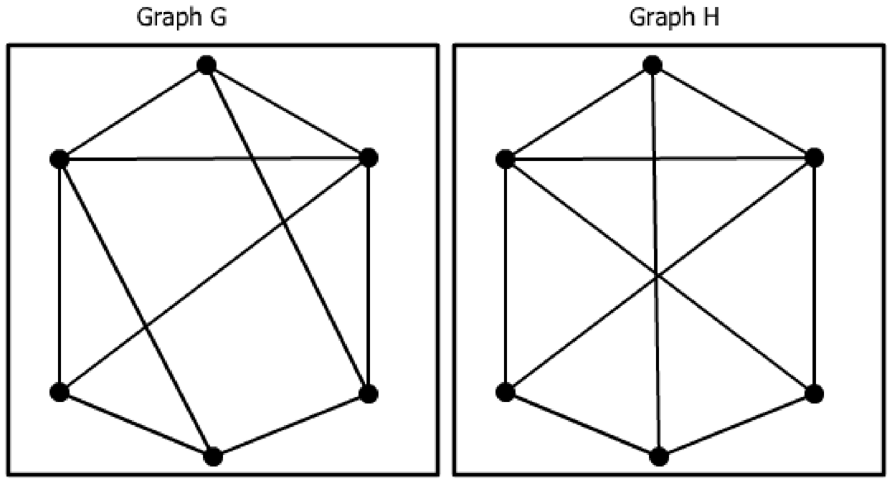
Latihan

- Tentukan apakah pasangan graf berikut bersifat isomorfik.



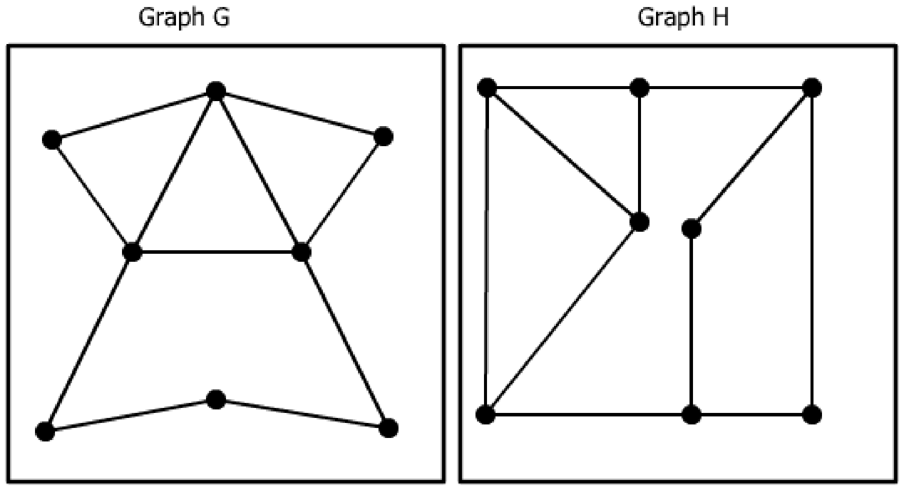
Latihan

- Tentukan apakah pasangan graf berikut bersifat isomorfik.



Latihan

- Tentukan apakah pasangan graf berikut bersifat isomorfik.



Contoh pemanfaatan isomorfisme

- Dalam bidang *bioinformatics*
  - Graf molecular dapat memodelkan senyawa kimia di mana atom sebagai vertex dan ikatan kimia antar atom sebagai sisi
  - Ketika suatu senyawa kimia baru dapat disintesa maka dapat dibandingkan dengan basis data senyawa yang sudah pernah ada
- Dalam bidang *electronics*
  - Sirkuit elektronik dapat dimodelkan dengan graf di mana komponen elektronik sebagai vertex dan hubungan antar komponen sebagai sisi
  - Isomorfisme dapat digunakan untuk menentukan apakah sirkuit yang dibuat sesuai dengan model awal
  - Isomorfisme dapat juga digunakan untuk menentukan apakah produk sirkuit perusahaan lain melanggar paten

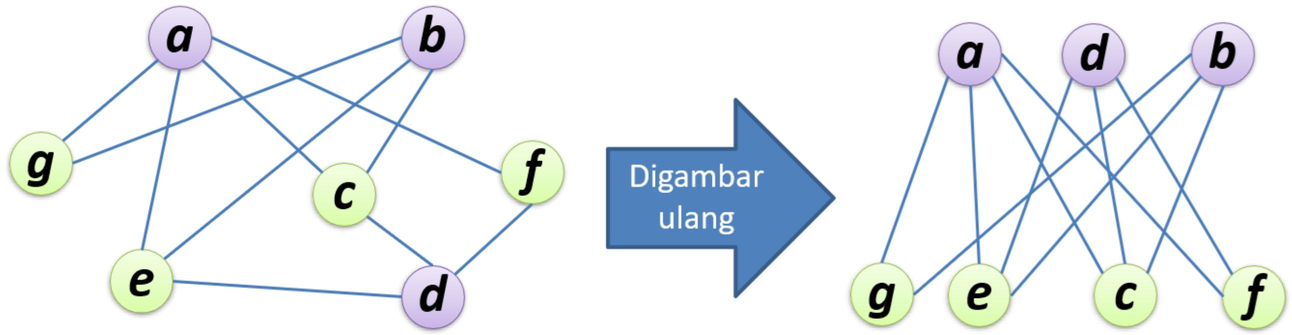
Bipartite dan Matching

Graf Bipartite

- Definisi

Sebuah graf G = (V, E) disebut graf *bipartite* jika dan hanya jika V dapat dipartisi menjadi dua himpunan tidak kosong V_1 dan V_2 yang saling lepas sedemikian sehingga untuk setiap sisi $e \in E$, berlaku salah satu *endpoint* e ada di dalam V_1 dan *endpoint* lainnya ada di dalam V_2.

- Contoh
  - Graf G = (V, E) dengan V = { a, b, c, d, e, f, g } dan E = { (a, c), (a, e), (a, f), (a, g), (b, c), (b, e), (b, g), (d, c), (d, e), (d, f), }
    - Terdapat U = { a, b, d } dan W = { c, e, f, g }



Graf Bipartite Lengkap

- Definisi

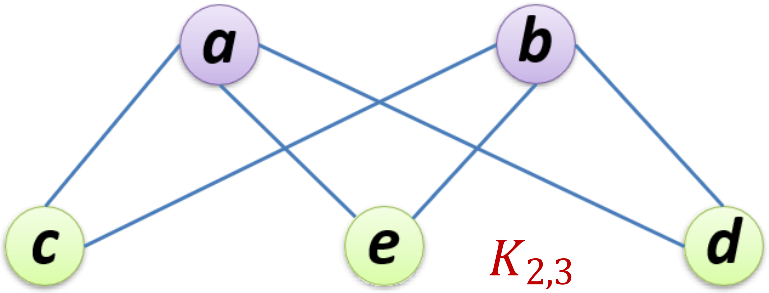
Sebuah graf $G = (V, E)$ disebut graf bipartite lengkap, $K_{m,n}$, apabila V merupakan gabungan dari 2 (dua) buah subset yaitu:

– U dengan $m \neq 0$ unsur

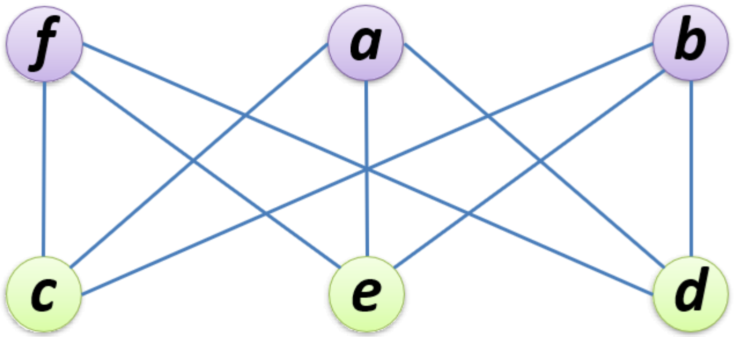
– W dengan $n \neq 0$ unsur yang saling lepas,

sedemikian sehingga sisi $(u, w) \in E \Leftrightarrow u \in U$ dan $w \in W$

- Contoh



$K_{2,3}$



$K_{3,3}$

Menentukan Bipartite

- Definisi

Sebuah graf sederhana adalah bipartite jika dan hanya jika setiap simpulnya bisa diberi satu dari dua warna berbeda sedemikian sehingga tidak ada simpul yang saling bertetangga yang memiliki warna yang sama.

Matching

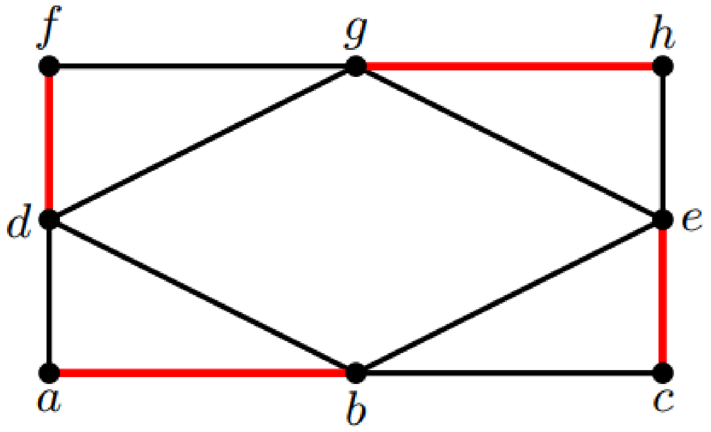
- Definisi

Diketahui $G = (V, E)$ adalah sebuah graf tidak berarah. Sebuah *matching* adalah himpunan sisi $M \subseteq E$ sedemikian sehingga tidak ada dua sisi M yang bertumpuan pada vertex yang sama.

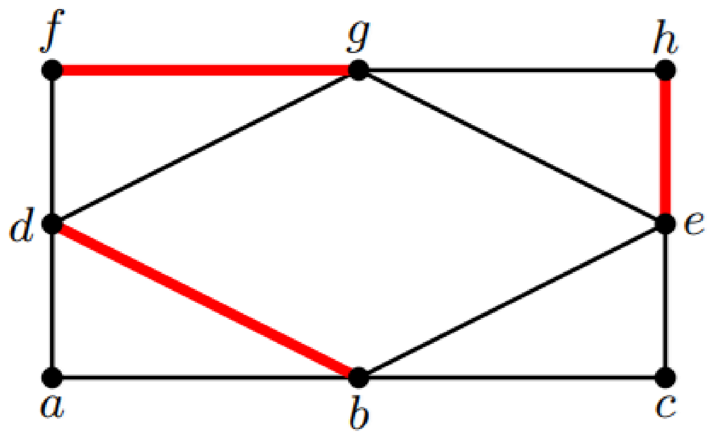
- *Matching* M disebut *maximal* jika M bukan proper subset dari *matching* lainnya pada G
- *Matching* M disebut *maximum* jika kardinalitas M lebih besar atau sama dengan kardinalitas *matching* lainnya pada G
- *Matching* M disebut *perfect* jika semua vertex pada G memiliki pasangan
- Jika G adalah graf bipartite dengan V_1 dan V_2 merupakan partisinya, maka *matching* M dikatakan lengkap (*complete*) dari V_1 ke V_2 jika setiap vertex di V_1 memiliki pasangan di V_2.

Contoh

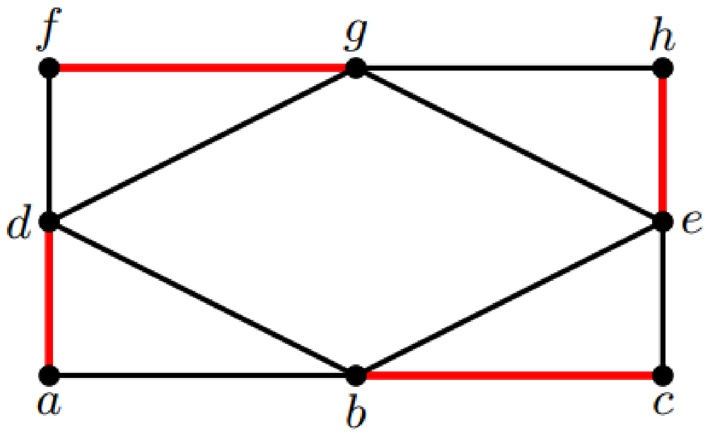
Perfect (and maximum) matching of size 4



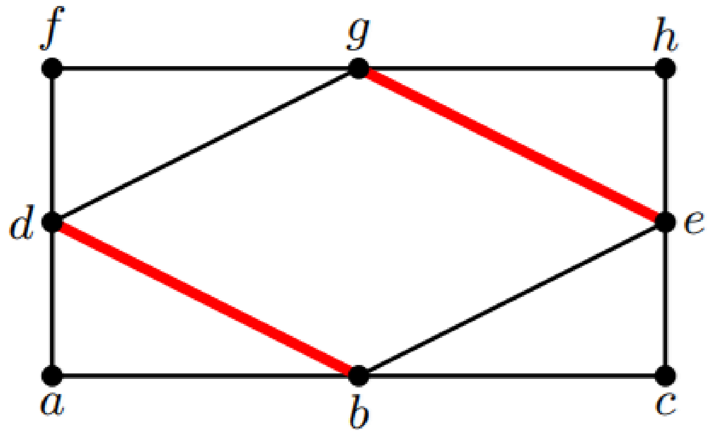
Maximal matching of size 3



Perfect (and maximum) matching of size 4



Maximal matching of size 2



Lintasan Terpendek

Lintasan Terpendek

Contoh Aplikasi

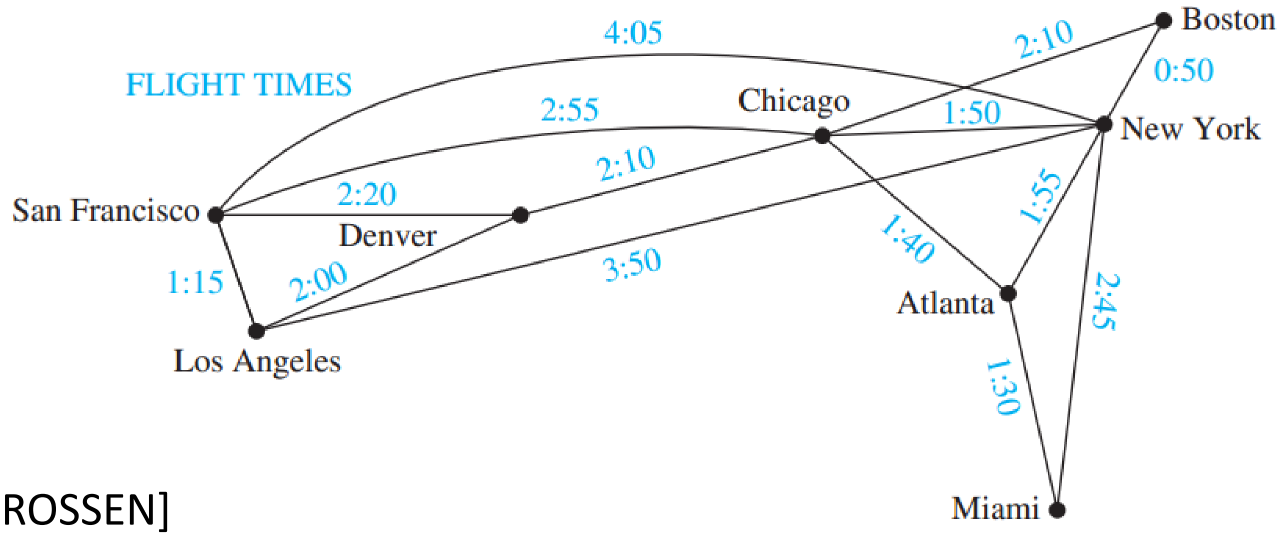
Kasus sistem penerbangan, setiap kota merupakan *vertex* dari graf, jalur penerbangan antar kota merupakan *edge* dari graf.

Berapa waktu minimal yang diperlukan untuk mengunjungi kota tertentu?

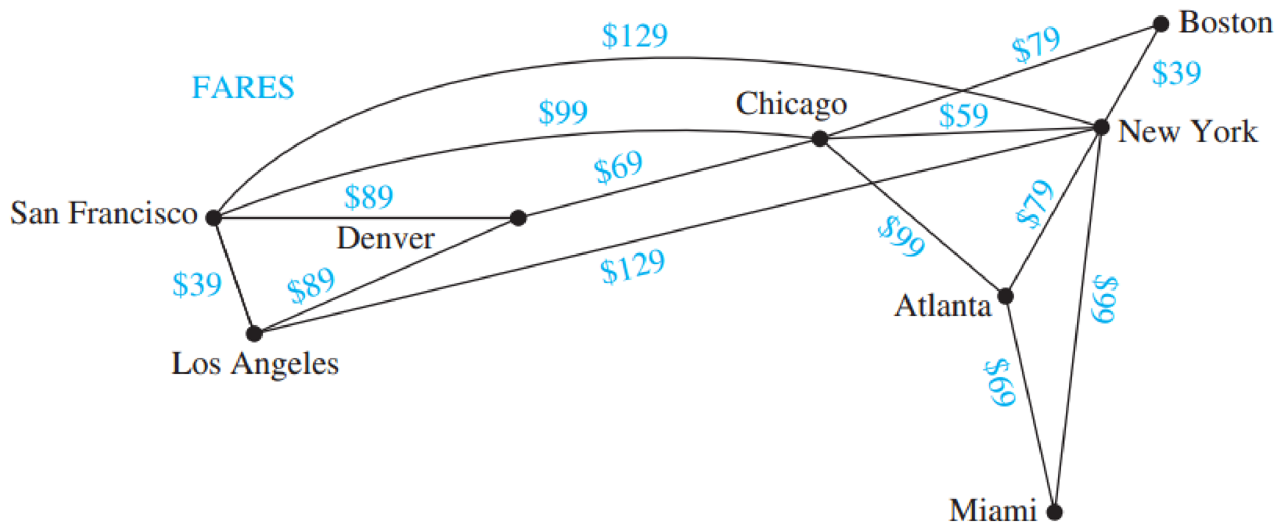
Berapa ongkos minimal yang diperlukan untuk mengunjungi kota tertentu?

Untuk menerapkan teori graf pada permasalahan seperti ini dibutuhkan pengertian tentang graf berbobot

Lintasan Terpendek



[ROSSEN]



Lintasan Terpendek

Definisi Graf Berbobot

Sebuah graf $G = (V, E)$ disebut graf berbobot (*weighted graph*) apabila terdapat fungsi bobot bernilai real W pada himpunan E, $W: E \rightarrow R$

Nilai $W(e)$ disebut bobot untuk sisi e, $\forall e \in E$

Graf berbobot dinyatakan sebagai $G = (V, E, W)$

Lintasan Terpendek

- Contoh graf berbobot
  - Graf representasi jaringan penerbangan udara
    - *V* = himpunan kota
    - *E* = himpunan rute penerbangan antar kota pada *V*
    - *W* = himpunan ongkos penerbangan tiap rute di *E* atau fungsi bernilai riil pada *E* yang menyatakan ongkos
  - Graf representasi jaringan komputer
    - *V* = himpunan computer
    - *E* = himpunan jalur kabel langsung antar dua computer
    - *W* = fungsi bernilai riil pada *E* yang menyatakan jarak atau ongkos atau waktu

Lintasan Terpendek

- Permasalahan lintasan terpendek dapat berarti salah satu dari menentukan:
  - Jalur termurah
  - Jalur terdekat
  - Jalur tercepat
- Makna dari lintasan terpendek bergantung pada
  - Makna fungsi bobot yang terdapat pada graf yang dibentuk

Lintasan Terpendek: Algoritma Dijkstra

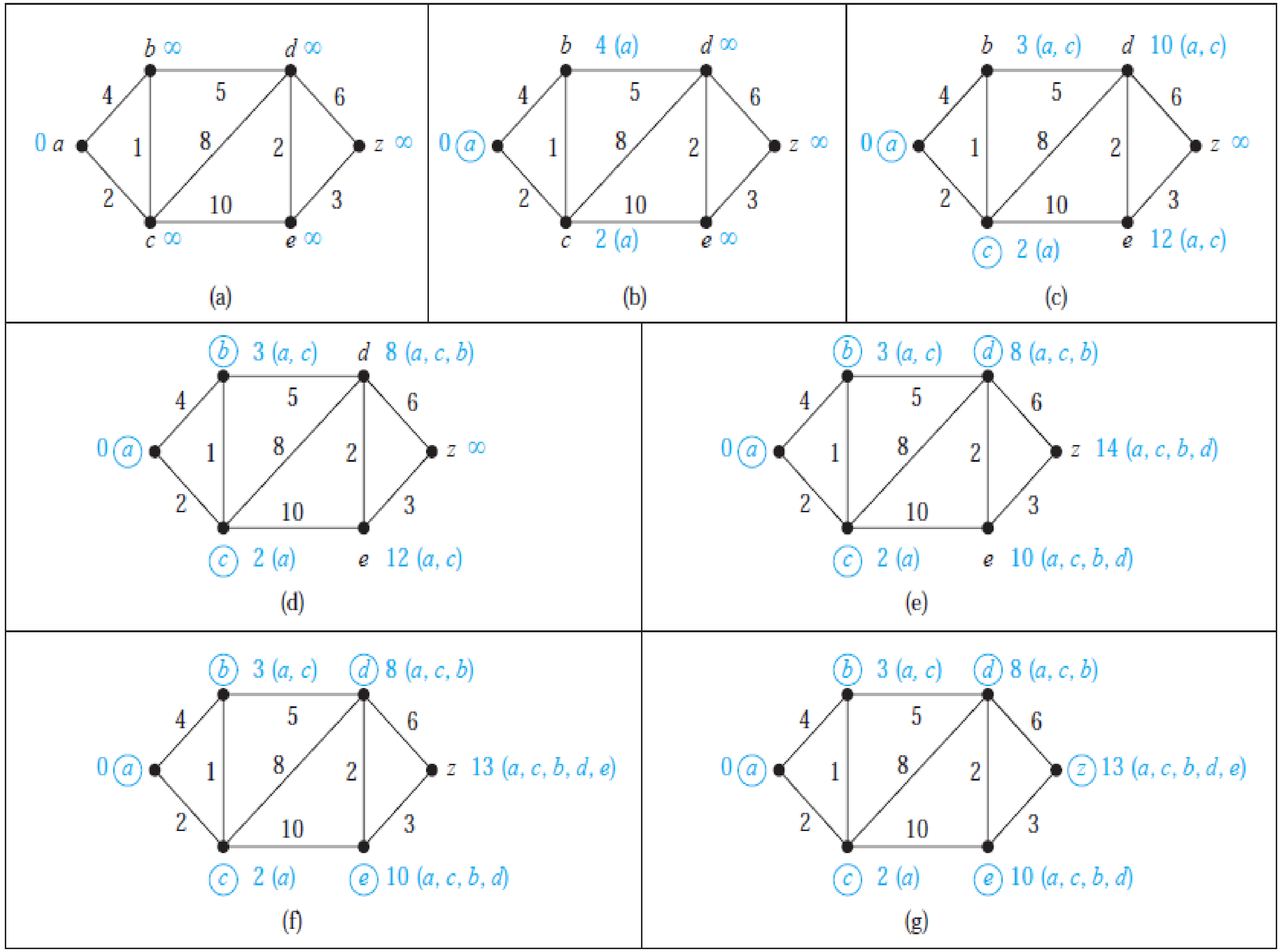
- Algoritma Dijkstra
  - Merupakan salah satu cara yang dapat digunakan untuk menyelesaikan permasalahan menemukan lintasan terpendek pada suatu graf berbobot
  - Inti dari algoritma Dijkstra:
    - Misalkan ingin dicari lintasan terpendek $P(a, z)$ dari vertex $a \in V$ dan vertex $z \in V$
    - Dicari dahulu lintasan terpendek $P(a, b)$ dari vertex a ke suatu vertex b di V
    - Selanjutnya dicari lintasan terpendek $P(a, c)$ dari vertex a ke suatu vertex c di V
    - Proses dilanjutkan terus menerus dan berhenti ketika lintasan terpendek $P(a, z)$ diperoleh

Lintasan Terpendek: Algoritma Dijkstra

ALGORITHM 1 Dijkstra's Algorithm.

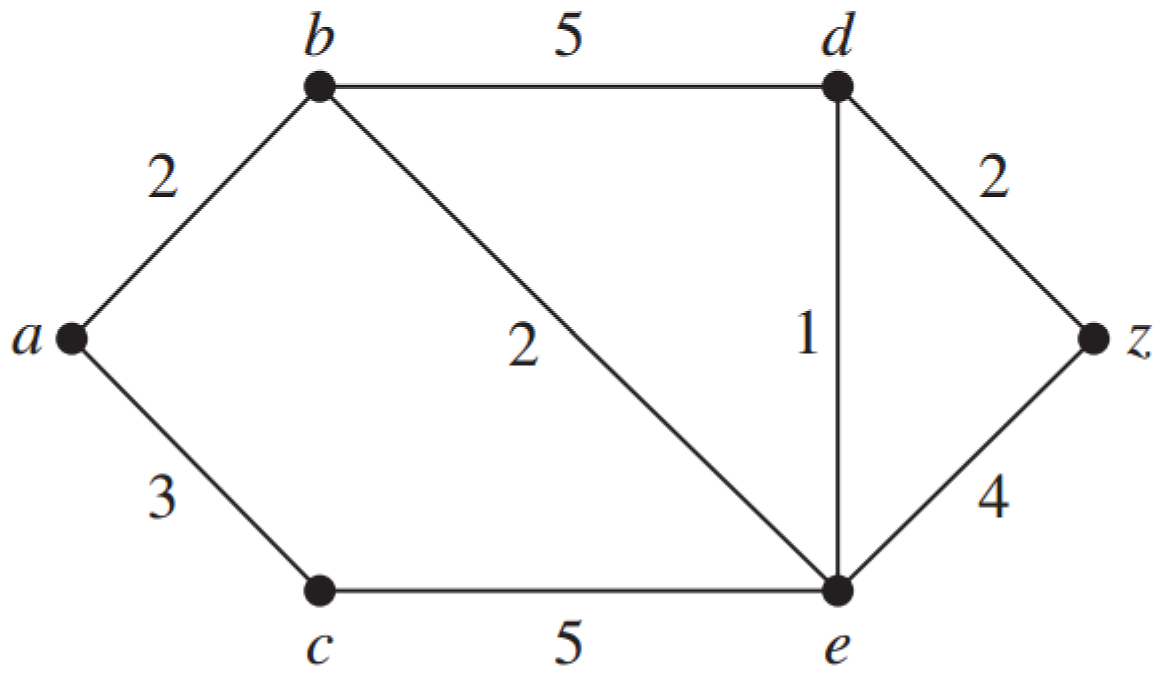
procedure *Dijkstra*(G: weighted connected simple graph, with
  all weights positive)
{G has vertices $a = v_0, v_1, \ldots, v_n = z$ and lengths $w(v_i, v_j)$
  where $w(v_i, v_j) = \infty$ if $\{v_i, v_j\}$ is not an edge in G}
for $i := 1$ **to** n
  $L(v_i) := \infty$
$L(a) := 0$
$S := \emptyset$
{the labels are now initialized so that the label of a is 0 and all
  other labels are ∞, and S is the empty set}
while $z \notin S$
  $u :=$ a vertex not in S with $L(u)$ minimal
  $S := S \cup \{u\}$
  for all vertices v not in S
    if $L(u) + w(u, v) < L(v)$ **then** $L(v) := L(u) + w(u, v)$
    {this adds a vertex to S with minimal label and updates the
    labels of vertices not in S}
return $L(z)$ {$L(z) =$ length of a shortest path from a to z}

Lintasan Terpendek: Algoritma Dijkstra



Lintasan Terpendek

- Carilah lintasan terpendek pada graf berbobot berikut ini!



Selamat belajar…