# Graph (Contd) :Aplikasi Graph

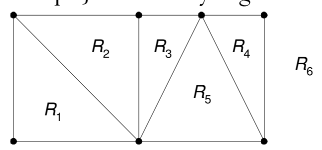
Sesi 11



## Graf Planar (*Planar Graph*) dan Graf Bidang (*Plane Graph*) -contd

- Rumus Euler : $n - e + f = 2$

  dimana
  - $f$ = jumlah wilayah
  - $e$ = jumlah sisi
  - $n$ = jumlah simpul
- Ex: Berapa jumlah wilayah graf berikut ini?

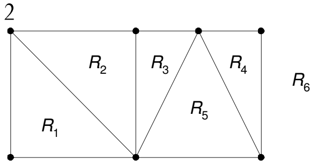



  - *Solusi:* $e = 11$ dan $n = 7$, maka $f = 11 - 7 + 2 = 6$



## Graf Planar (*Planar Graph*) dan Graf Bidang (*Plane Graph*) -contd

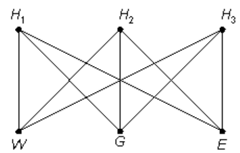
- Pada graf planar sederhana terhubung dengan $f$ wilayah, $n$ buah simpul, dan $e$ buah sisi (dengan $e > 2$) selalu berlaku ketidaksamaan berikut:
  - $e \geq 3f/2$
  - $e \leq 3n - 6$
- Ex: Graf berikut ini adalah graf planar kerena $6 \geq 3(4)/2$ dan $6 \leq 3(4) - 2$






## Graf Planar (*Planar Graph*) dan Graf Bidang (*Plane Graph*) -contd

- Ex: Bagaimana dengan graf berikut ini




  $e = 9$, $n = 6$ sehingga $9 \leq (3)(6) - 6 = 12$ (benar, $e \leq 3n - 6$) padahal graf *tersebut* bukan graf planar!
- Untuk mengatasi hal ini, buat asumsi baru: setiap daerah pada graf planar dibatasi oleh paling sedikit empat buah sisi. Dari penurunan rumus diperoleh

$$e \leq 2n - 4$$

- Sehingga $9 \leq (2)(6) - 4 = 8$ (salah) yang berarti *graf tersebut* bukan graf planar.

## Lintasan dan Sirkuit Euler

- **Lintasan Euler** ialah lintasan yang melalui masing-masing sisi di dalam graf tepat satu kali. Syarat:
  - Graf terhubung
  - Memiliki dua buah simpul berderajat ganjil atau tidak ada simpul yang berderajat ganjil
- **Sirkuit Euler** ialah sirkuit yang melewati masing-masing sisi tepat satu kali. Syarat:
  - Graf terhubung
  - Semua simpul pada graf berderajat genap
- Graf yang mempunyai sirkuit Euler disebut **graf Euler** (*Eulerian graph*). Graf yang mempunyai lintasan Euler dinamakan juga graf **semi-Euler** (*semi-Eulerian graph*)

5

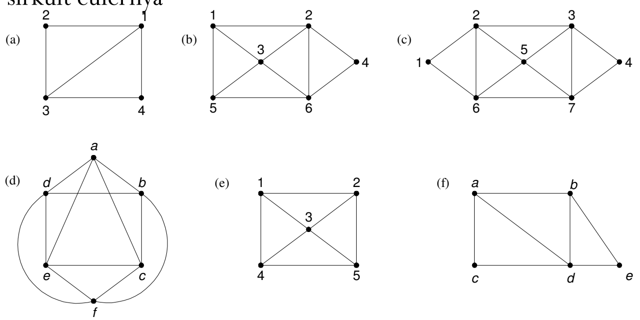
## Teorema Lintasan dan Sirkuit Euler

1. Graf tidak berarah memiliki lintasan Euler jika dan hanya jika terhubung dan memiliki dua buah simpul berderajat ganjil atau tidak ada simpul berderajat ganjil sama sekali
2. Graf tidak berarah *G* adalah graf Euler (memiliki sirkuit Euler) jika dan hanya jika setiap simpul berderajat genap. (Note: graf yang memiliki sirkuit Euler pasti mempunyai lintasan Euler, tetapi tidak sebaliknya)
3. Graf berarah *G* memiliki sirkuit Euler jika dan hanya jika *G* terhubung dan setiap simpul memiliki derajat-masuk dan derajat-keluar sama. *G* memiliki lintasan Euler jika dan hanya jika *G* terhubung dan setiap simpul memiliki derajat-masuk dan derajat-keluar sama kecuali dua simpul, yang pertama memiliki derajat-keluar satu lebih besar derajat-masuk, dan yang kedua memiliki derajat-masuk satu lebih besar dari derajat-keluar

6

## Studi Kasus Lintasan dan Sirkuit Euler

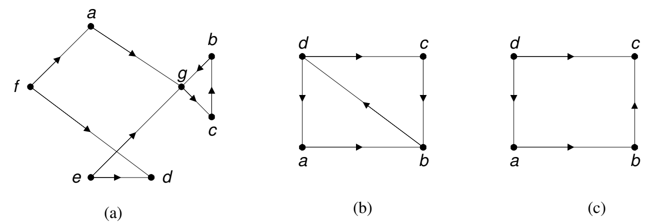
- Manakah dari graf-graf tidak berarah berikut ini yang memiliki lintasan dan/atau sirkuit euler. Jika ada tulis lintasan dan/atau sirkuit eulernya




7

## Studi Kasus Lintasan dan Sirkuit Euler

- Manakah dari graf-graf berarah berikut ini yang memiliki lintasan dan/atau sirkuit euler. Jika ada tulis lintasan dan/atau sirkuit eulernya




8

## Lintasan dan Sirkuit Hamilton

- **Lintasan Hamilton** ialah lintasan yang melalui tiap simpul di dalam graf tepat satu kali.
- **Sirkuit Hamilton** ialah sirkuit yang melalui tiap simpul di dalam graf tepat satu kali, kecuali simpul asal (sekaligus simpul akhir) yang dilalui dua kali.
- Graf yang memiliki sirkuit Hamilton dinamakan **graf Hamilton**, sedangkan graf yang hanya memiliki lintasan Hamilton disebut **graf semi-Hamilton**

9

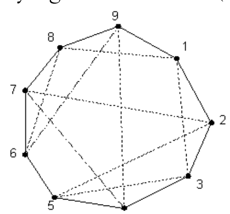
## Teorema Lintasan dan Sirkuit Hamilton

1. Syarat cukup (jadi bukan syarat perlu) supaya graf sederhana $G$ dengan $n$ ($\geq 3$) buah simpul adalah graf Hamilton ialah bila derajat tiap simpul paling sedikit $n/2$ (yaitu, $d(v) \geq n/2$ untuk setiap simpul $v$ di $G$)
2. Setiap graf lengkap adalah graf Hamilton
3. Di dalam graf lengkap $G$ dengan $n$ buah simpul ($n \geq 3$), terdapat $(n - 1)!/2$ buah sirkuit Hamilton.
4. Di dalam graf lengkap $G$ dengan $n$ buah simpul ($n \geq 3$ dan n ganjil), terdapat $(n - 1)/2$ buah sirkuit Hamilton yang saling lepas (tidak ada sisi yang beririsan). Jika $n$ genap dan $n \geq 4$, maka di dalam G terdapat $(n - 2)/2$ buah sirkuit Hamilton yang saling lepas

10

## Studi Kasus Lintasan dan Sirkuit Hamilton

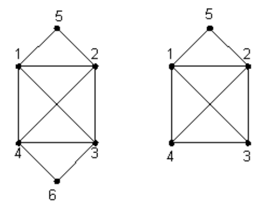
- Sembilan anggota sebuah klub bertemu tiap hari untuk makan siang pada sebuah meja bundar. Mereka memutuskan duduk sedemikian sehingga setiap anggota mempunyai tetangga duduk berbeda pada setiap makan siang. Berapa hari pengaturan tersebut dapat dilaksanakan?
- Solusi : berdasarkan teorema keempat maka jumlah pengaturan tempat duduk yang berbeda adalah $(9 - 1)/2 = 4$




11

## Studi Kasus Lintasan dan Sirkuit Hamilton

- Beberapa graf dapat mengandung sirkuit Euler dan sirkuit Hamilton sekaligus, mengandung sirkuit Euler tetapi tidak mengandung sirkuit Hamilton, mengandung sirkuit Euler dan lintasan Hamilton, mengandung lintsan Euler maupun lintasan Hamilton, tidak mengandung lintasan Euler namun mengandung sirkuit Hamilton, dan sebagainya. Dari graf-graf berikut ini manakah yang memenuhi pernyataan tersebut

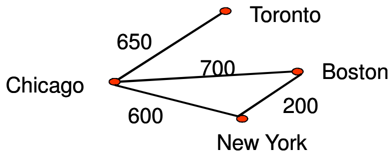



(a) (b)

12

## Shortest Path Problems

- We can assign weights to the edges of graphs, for example to represent the distance between cities in a railway network:




- Such weighted graphs can also be used to model computer networks with response times or costs as weights.
- One of the most interesting questions that we can investigate with such graphs is: What is the **shortest path** between two vertices in the graph, that is, the path with the **minimal sum of weights** along the way?
- This corresponds to the shortest train connection or the fastest connection in a computer network

13

## Dijkstra's Algorithm

- **Theorem:** Dijkstra's algorithm finds the length of a shortest path between two vertices in a connected simple undirected weighted graph.
  - Dijkstra's algorithm is an iterative procedure that finds the shortest path between to vertices a and z in a weighted graph.
  - It proceeds by finding the length of the shortest path from a to successive vertices and adding these vertices to a distinguished set of vertices S.
  - The algorithm terminates once it reaches the vertex z.
- **Theorem:** Dijkstra's algorithm uses **O(n$^2$)** operations (additions and comparisons) to find the length of the shortest path between two vertices in a connected simple undirected weighted graph

14

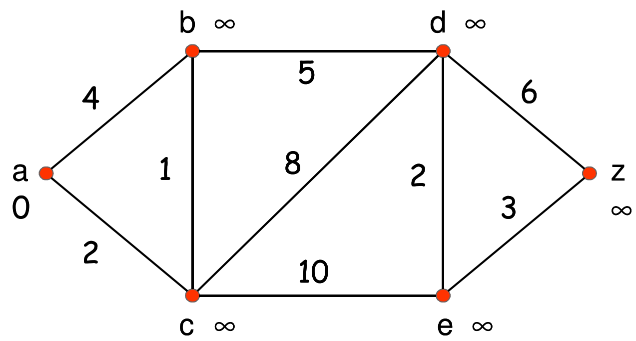
## Dijkstra's Algorithm

```
procedure DIJKSTRA_SINGLE_SOURCE_SP(V, E, w, s)
begin
    V_T := {s};
    for all v ∈ (V − V_T) do
        if (s, v) exists set l[v] := w(s, v);
        else set l[v] := ∞;
    while V_T ≠ V do
    begin
        find a vertex u such that l[u] := min{l[v]|v ∈ (V − V_T)};
        V_T := V_T ∪ {u};
        for all v ∈ (V − V_T) do
            l[v] := min{l[v], l[u] + w(u, v)};
    endwhile
end DIJKSTRA_SINGLE_SOURCE_SP
```

**Algorithm 10.2** Dijkstra's sequential single-source shortest paths algorithm.

XI - 15

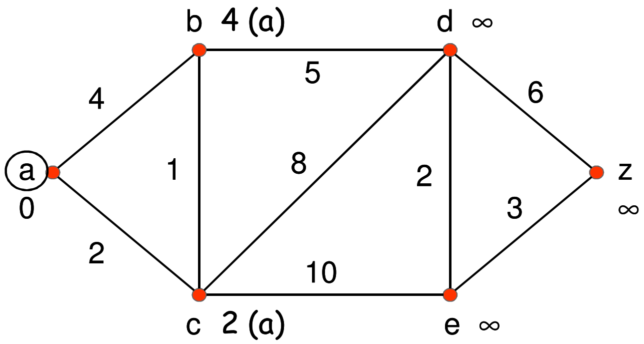
## Example Dijkstra's Algorithm




**Step 0**

XI - 16

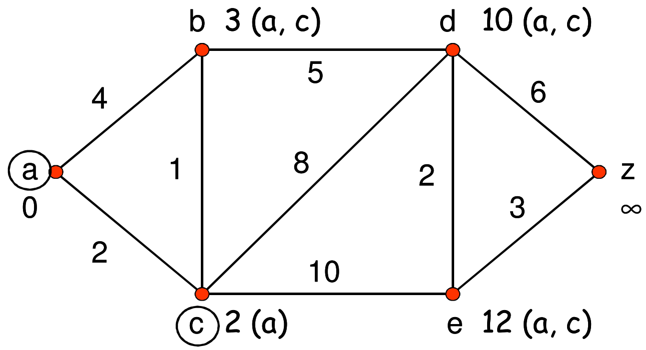
## Example Dijkstra's Algorithm




**Step 1**

XI - 17

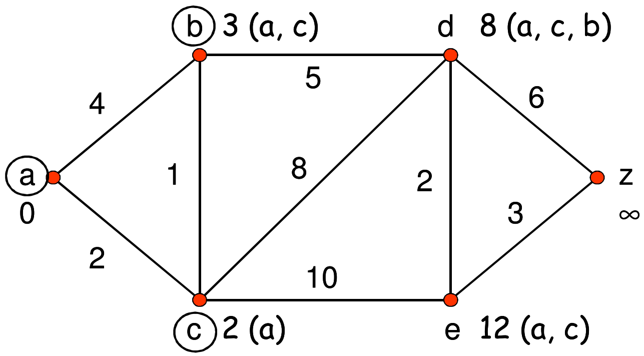
## Example Dijkstra's Algorithm




**Step 2**

XI - 18

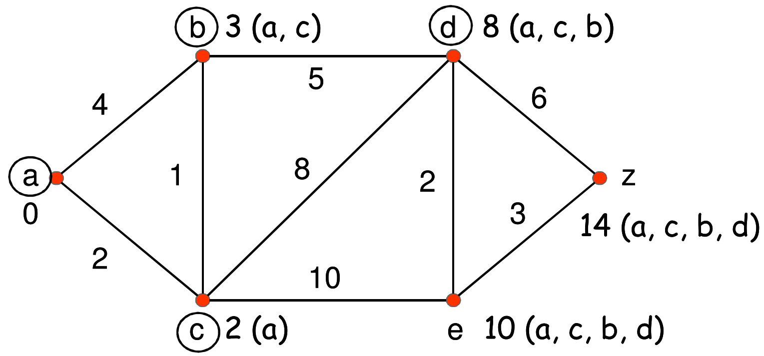
## Example Dijkstra's Algorithm




**Step 3**

XI - 19

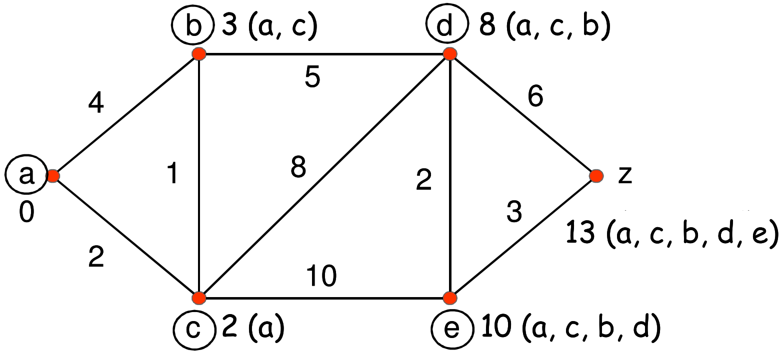
## Example Dijkstra's Algorithm




**Step 4**

XI - 20

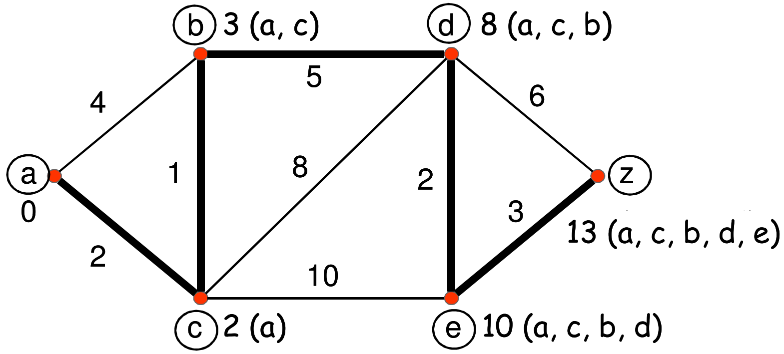
## Example Dijkstra's Algorithm




**Step 5**

XI - 21

## Example Dijkstra's Algorithm

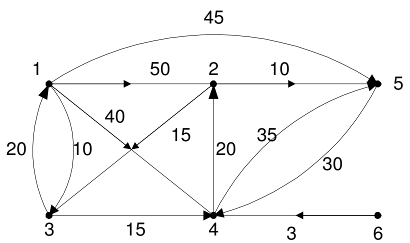



**Step 6**

XI - 22

## Studi Kasus Dijkstra's Algorithm (1)

- Tentukan lintasan terpendek dari simpul 1 ke simpul yang lain dari graf berikut ini




23

## Studi Kasus Dijkstra's Algorithm (2)

- Solusi :

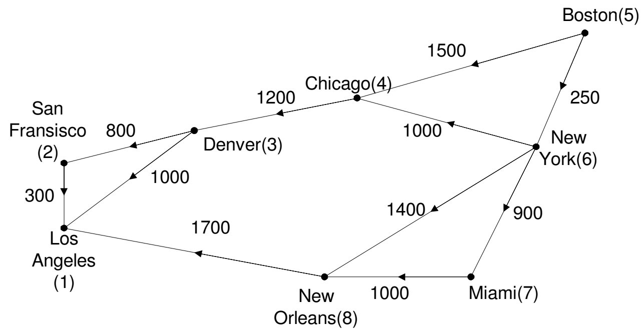
| Lelaran | Simpul yang dipilih | Lintasan | S 1 | S 2 | S 3 | S 4 | S 5 | S 6 | D 1 | D 2 | D 3 | D 4 | D 5 | D 6 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Inisial | - | - | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 50 (1,2) | 10 (1,3) | 40 (1,4) | 45 (1,5) | ∞ (1,6) |
| 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | ∞ | 50 (1,2) | 10 (1,3) | 40 (1,4) | 45 (1,5) | ∞ (1,6) |
| 2 | 3 | 1, 3 | 1 | 0 | 1 | 0 | 0 | 0 | ∞ | 50 (1,2) | 10 (1,3) | 25 (1,3,4) | 45 (1,5) | ∞ (1,6) |
| 3 | 4 | 1, 3, 4 | 1 | 0 | 1 | 1 | 0 | 0 | ∞ | 45 (1,3,4,2) | 10 (1,3) | 25 (1,3,4) | 45 (1,4) | ∞ (1,6) |
| 4 | 2 | 1, 3, 4, 2 | 1 | 1 | 1 | 1 | 0 | 0 | ∞ | 45 (1,3,4,2) | 10 (1,3) | 25 (1,3,4) | 45 (1,4) | ∞ (1,6) |
| 5 | 5 | 1, 5 | 1 | 1 | 1 | 1 | 1 | 0 | ∞ | 45 | 10 | 25 | 45 | ∞ |

Jadi, lintasan terpendek dari:
1 ke 3 adalah 1, 3 dengan panjang = 10
1 ke 4 adalah 1, 3, 4 dengan jarak = 25
1 ke 2 adalah 1, 3, 4, 2 dengan jarak = 45
1 ke 5 adalah 1, 5 dengan jarak = 45
1 ke 6 tidak ada

24

## Studi Kasus Dijkstra's Algorithm (5)

- Tentukan lintasan terpendek dari simpul 5 ke simpul yang lain dari graf berikut ini



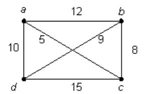



## The Traveling Salesman Problem

- The **traveling salesman problem** is one of the classical problems in computer science.
- A traveling salesman wants to visit a number of cities and then return to his starting point. Of course he wants to save time and energy, so he wants to **determine the shortest path** for his trip.
- We can represent the cities and the distances between them by a weighted, complete, undirected graph.
- The problem then is to find the **circuit of minimum total weight that visits each vertex exactly one** ➔ <u>menentukan sirkuit Hamilton yang memiliki bobot minimum</u>

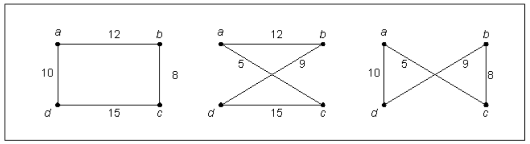


## Ex: Traveling Salesman Problem




Graf di atas memiliki (4 – 1)!/2 = 3 sirkuit Hamilton, yaitu:

$I_1$ = (*a*, *b*, *c*, *d*, *a*) atau (*a*, *d*, *c*, *b*, *a*) ==> panjang = 10 + 12 + 8 + 15 = 45
$I_2$ = (*a*, *c*, *d*, *b*, *a*) atau (*a*, *b*, *d*, *c*, *a*) ==> panjang = 12 + 5 + 9 + 15 = 41
$I_3$ = (*a*, *c*, *b*, *d*, a) atau (*a*, *d*, *b*, *c*, *a*) ==> panjang = 10 + 5 + 9 + 8 = 32

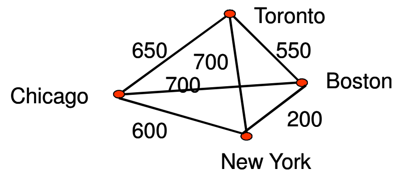



Jadi, sirkuit Hamilton terpendek adalah $I_3$ = (*a*, *c*, *b*, *d*, *a*) atau (*a*, *d*, *b*, *c*, *a*) dengan panjang sirkuit = 10 + 5 + 9 + 8 = 32.



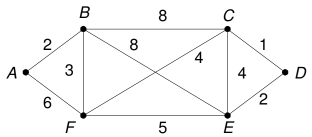
## The Traveling Salesman Problem

- **Example:** What path would the traveling salesman take to visit the following cities?



## The *Chinese Postman Problem*

- Dikemukakan oleh Mei Gan (berasal dari Cina) pada tahun 1962.
- Masalahnya adalah sebagai berikut: *seorang tukang pos akan mengantar surat ke alamat-alamat sepanjang jalan di suatu daerah. Bagaimana ia merencanakan rute perjalanannya supaya ia melewati setiap jalan tepat sekali dan kembali lagi ke tempat awal keberangkatan.*
  - Solusi ➔ menentukan sirkuit Euler di dalam graf
- Ex:




- Lintasan yang dilalui tukang pos: *A*, *B*, *C*, *D*, *E*, *F*, *C*, *E*, *B*, *F*, *A*

29

## Referensi

1. Rinaldi Munir, "Materi Kuliah Matematika Diskrit",Informatika-ITB, Bandung,2003
2. Rinaldi Munir, "Matematika Diskrit",Informatika, Bandung,2001

30