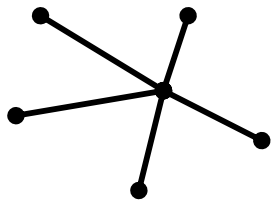
ทฤษฎีกราฟ (Graph Theory)

ทฤษฎีกราฟ

- ทฤษฎีกราฟถูกนำมาใช้แก้ปัญหา เพื่อช่วยให้เห็นภาพพจน์ของการเชื่อมโยง หรือความสัมพันธ์มากขึ้น เช่น ระยะทาง เส้นทาง วงจร เป็นต้น
- ตัวอย่างเช่น กราฟนำมาใช้แสดงการเชื่อมต่อคอมพิวเตอร์ในระบบเครือข่ายได้ โทโปโลยี (Topology) ต่างๆ ในระบบเครือข่าย

โทโปโลยีระบบเครือข่าย

- (1) โทโปโลยีแบบดาว (Star Topology)

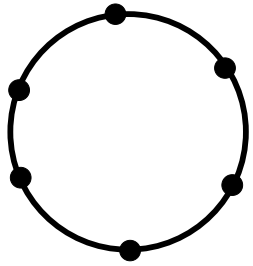


โทโปโลยีระบบเครือข่าย

- (2) โทโปโลยีแบบบัส

โทโปโลยีระบบเครือข่าย

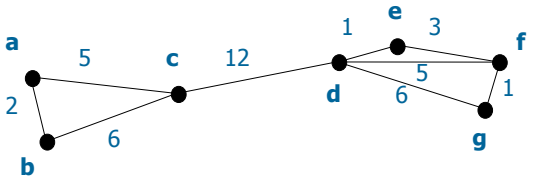
- (3) โทโปโลยีแบบริง



การแสดงแผนที่ด้วยกราฟ

- ตัวอย่างเช่น การสร้างแผนที่ เพื่อแสดงระยะทางจากจุดต่างๆ สถานที่ต่างๆ ในแผนที่แทนด้วยจุดและเส้นทางจากสถานที่หนึ่งไปยังอีกสถานที่หนึ่งแทนได้ด้วยขอบที่เชื่อมจุดต่างๆ เข้าด้วยกัน สำหรับระยะทางจากจุดหนึ่งไปยังอีกจุด ก็เขียนตัวเลขกำกับที่ขอบ

การแสดงแผนที่ด้วยกราฟ



- จากภาพตัวอย่าง จัดว่าเป็นการกำหนดหรือแสดงโครงสร้างแบบไม่ต่อเนื่องที่ประกอบด้วย
  - จุดยอด (*Vertices*)
  - ขอบ (*Edges*) : เชื่อมต่อระหว่างจุดยอด

ประเภทของกราฟ

- (1) กราฟไม่ระบุทิศทาง (Undirected graph)
  - กราฟอย่างง่าย (Simple graph)
  - มัลติกราฟ (Multigraph)
  - Pseudograph
- (2) กราฟระบุทิศทาง (Directed graph)
  - Directed graph
  - Directed Multigraph

กราฟไม่ระบุทิศทาง

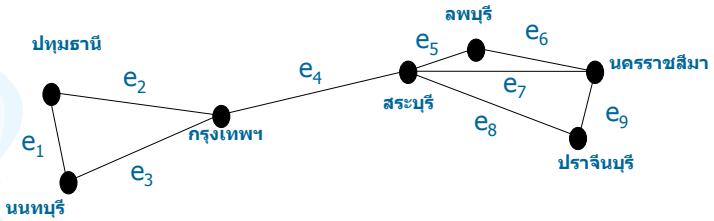
- **กราฟอย่างง่าย (Simple graph)**

นิยาม Simple graph $G = (V,E)$
โดยที่ V เป็นเซตของ *vertices* และ E เป็นเซตของเส้นที่เชื่อมต่อระหว่าง vertices ที่อยู่ในเซต V เรียกว่า *edges*
ฟังก์ชัน f เป็นฟังก์ชันจาก E ไปยัง $\{ \{u,v\} \mid u,v \in V \}$

ตัวอย่างกราฟอย่างง่าย

- เครือข่ายคอมพิวเตอร์ซึ่งใช้สายโทรศัพท์ในการเชื่อมต่อ 1 เส้นเชื่อมต่อเครื่องคอมพิวเตอร์ 2 เครื่องภายในเครือข่าย สายโทรศัพท์แต่ละเส้นสื่อสารแบบสองทิศทางและไม่มีเครื่องคอมพิวเตอร์มีเชื่อมต่อไปยังตัวเครื่องเอง ประกอบด้วย **จุดยอด (Vertices)** ซึ่งใช้แทนเครื่องคอมพิวเตอร์และ**ขอบ (edges)** ไม่มีระบุทิศทางใช้แทนสายโทรศัพท์ ซึ่งแต่ละขอบจะเชื่อมต่อระหว่าง 2 จุดยอดและมีเพียงขอบ (*edges*) เดียวระหว่างแต่ละจุดยอด

ตัวอย่างกราฟอย่างง่าย



เครือข่ายคอมพิวเตอร์แบบ Simple graph

ตัวอย่างกราฟอย่างง่าย

- จากกราฟตัวอย่างจะได้ G = (V,E) โดยที่
  - V = {กรุงเทพ , นนทบุรี , ปทุมธานี , สระบุรี , ลพบุรี , นครราชสีมา , ปราจีนบุรี }
  - E = { e1 , e2 , e3 , e4 , e5 , e6 , e7 , e8 , e9 }
  - ตัวอย่างฟังก์ชัน f จาก E ไปยัง V ดังนี้
  - f(e1) = {ปทุมธานี , นนทบุรี}
  - f(e2) = {ปทุมธานี , กรุงเทพฯ}
  - f(e3) = {กรุงเทพฯ , นนทบุรี}
  - ....

กราฟไม่ระบุทิศทาง

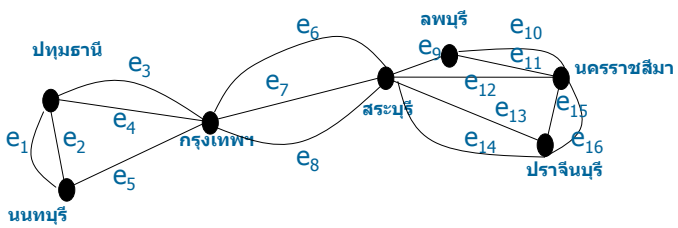
- **มัลติกราฟ (Multigraph)**

นิยาม Multigraph G = (V,E)
โดยที่ V เป็นเซตของ *vertices* และ E เป็นเซตของเส้นที่เชื่อมต่อระหว่าง vertices ที่อยู่ในเซต V เรียกว่า *edges*
ฟังก์ชัน *f* เป็นฟังก์ชันจาก E ไปยัง { {u,v} | u,v $\in$ V , u $\neq$ v } และ
ขอบ e_1 และ e_2 จะเรียกว่าเป็นขอบ multiple หรือ parallel เมื่อ $f(e_1) = f(e_2)$

ตัวอย่างมัลติกราฟ

- ในบางครั้งอาจจะต้องใช้สายโทรศัพท์หลายเส้นต่อระหว่างเครื่องคอมพิวเตอร์ เนื่องจากการสื่อสารในเครือข่ายคอมพิวเตอร์มีความแออัด ดังนั้นเครือข่ายใน simple graph ไม่เพียงพอในการใช้งาน จึงใช้ multigraphs แทน โดยที่ multigraphs ประกอบด้วยจุดยอดและขอบไม่ระบุทิศทางเชื่อมต่อจุดยอด และมีการเพิ่มขอบเชื่อมต่อระหว่างจุดยอด

ตัวอย่างมัลติกราฟ



เครือข่ายคอมพิวเตอร์แบบ Multigraph

ตัวอย่างมัลติกราฟ

- จากภาพสามารถยกตัวอย่างฟังก์ชัน f ที่ส่งค่า E ไปยัง V ได้ดังนี้
  - f(e1) = {ปทุมธานี , นนทบุรี} = f(e2)
  - f(e3) = f(e4) = {กรุงเทพ , ปทุมธานี}
  - f(e15) = f(e16) = {นครราชสีมา , ปราจีนบุรี}

กราฟไม่ระบุทิศทาง

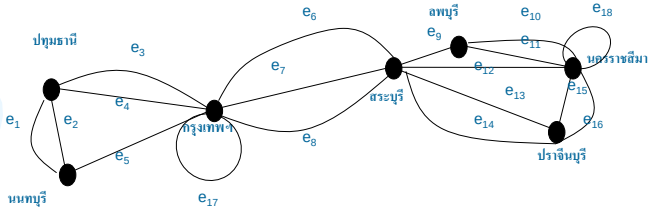
- **ซูโดกราฟ (Pseudograph)**

นิยาม Pseudograph G = (V,E)
โดยที่ V เป็นเซตของ *vertices* และ E เป็นเซตของเส้นที่เชื่อมต่อระหว่าง vertices ที่อยู่ในเซต V เรียกว่า *edges*
ฟังก์ชัน *f* เป็นฟังก์ชันจาก E ไปยัง { {u,v} | u,v ∈ V }
และ ขอบ(edges) จะเป็น loop ก็ต่อเมื่อ *f*(e) = {u} โดยที่ u ∈ V

ตัวอย่างซูโดกราฟ

- ในเครือข่ายคอมพิวเตอร์อาจจะประกอบด้วยสายโทรศัพท์ที่เชื่อมต่อเครื่องคอมพิวเตอร์ไปยังตัวเครื่องเอง ดังเครือข่ายที่แสดงในภาพด้านล่าง จึงไม่สามารถใช้ multigraph ในการสร้างแบบโครงสร้างของเครือข่ายได้ เนื่องจากมี loop ในเครือข่ายซึ่งเป็นขอบที่เชื่อมจากจุดยอดไปยังตัวเอง จึงใช้ Pseudograph แทน โดยที่ขอบของ Pseudograph อาจจะเชื่อมต่อจากจุดยอด 1 จุดไปยังจุดนั้นเอง

ตัวอย่างซูโดกราฟ



เครือข่ายคอมพิวเตอร์แบบ **Pseudograph**

ตัวอย่างซูโดกราฟ

- จากภาพสามารถยกตัวอย่างฟังก์ชัน f ที่ส่งค่า E ไปยัง V ที่มีลักษณะเป็น loop ได้ดังนี้
- f(e17) = {กรุงเทพฯ}
- f(e18) = {นครราชสีมา}

กราฟระบุทิศทาง

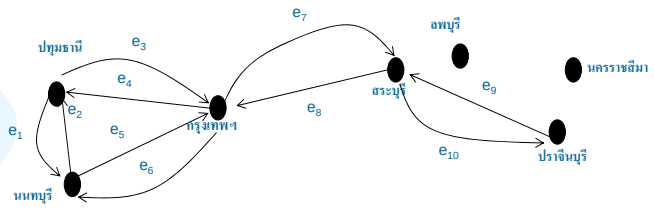
- **Directed graph** เป็นกราฟที่มีการระบุทิศทางของขอบ แสดงจุดยอดเริ่มต้นของขอบ และ จุดยอดปลายทางของขอบ

นิยาม **Directed graph G** คือกราฟที่ประกอบด้วย (V,E) โดยที่ V เป็นเซตของ *vertices* และ E เป็นเซตของขอบ (edges) ที่เป็นคู่อันดับของสมาชิกใน ฟังก์ชัน *f* เป็นฟังก์ชันจาก E ไปยัง { (u,v) | u,v ∈ V }

ตัวอย่าง Directed graph

- ในบางเหตุการณ์เครื่องคอมพิวเตอร์บางเครื่องไม่จำเป็นต้องการติดต่อสื่อสารแบบสองทิศทาง ดังตัวอย่างในภาพด้านล่างจะเห็นว่าเครื่องคอมพิวเตอร์ที่อยู่กรุงเทพฯ สามารถรับข้อมูลได้เพียงอย่างเดียวและไม่สามารถส่งข้อมูลได้
- ขอบ (edges) ของกราฟระบุทิศทางเป็นคู่อันดับของจุดยอด Loop เป็นคู่อันดับที่มีสมาชิกเหมือนกัน แต่จะไม่มีขอบ หลายขอบในทิศทางเดียวกันที่เชื่อมต่อระหว่างจุดยอด

ตัวอย่าง Directed graph



เครือข่ายคอมพิวเตอร์แบบ Directed graph

ตัวอย่าง Directed graph

- จากภาพสามารถยกตัวอย่างฟังก์ชัน f ที่ส่งค่า E ไปยัง V ได้ดังนี้
  - f(e1) = (ปทุมธานี , นนทบุรี)
  - f(e2) = (นนทบุรี , ปทุมธานี)
  - f(e3) = (ปทุมธานี , กรุงเทพฯ)
  - f(e4) = (กรุงเทพฯ , ปทุมธานี)

กราฟระบุทิศทาง

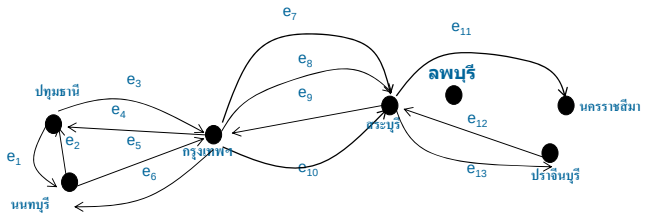
- **Directed Multigraph**

นิยาม Directed Multigraph คือกราฟที่ประกอบด้วย (V,E)
โดยที่ V เป็นเซตของ *vertices* และ E เป็นเซตของขอบ (edges) และ
ฟังก์ชัน f เป็นฟังก์ชันจาก E ไปยัง { (u,v) | u,v ∈ V }
โดยที่ e_1 และ e_2 จะเป็น Multiple edges เมื่อ $f(e_1) = f(e_2)$

ตัวอย่าง Directed Multigraph

- ในบางเหตุการณ์อาจจะใช้หลายเส้นทางการสื่อสาร โดยที่แต่ละเส้นทางสามารถสื่อสารได้เพียงทางเดียว เพื่อเพิ่มเส้นทางการสื่อสาร จึงมีการใช้ Directed Multigraph แทน Directed graph เนื่องจาก Directed Multigraph สามารถมีขอบระบุทิศทางได้หลายขอบขึ้นอยู่กับความต้องการ

ตัวอย่าง Directed Multigraph



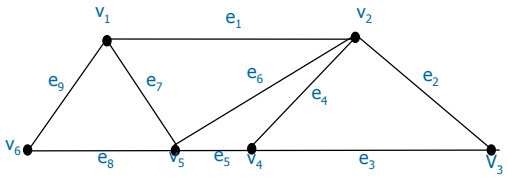
เครือข่ายคอมพิวเตอร์แบบ Multigraphs One-way lines

ตัวอย่าง Directed Multigraph

- จากภาพสามารถยกตัวอย่างฟังก์ชัน f ที่ส่งค่า E ไปยัง V ได้ดังนี้
  - f(e7) = (กรุงเทพ , สระบุรี) = f(e8) = f(e10)

ตัวอย่าง 1

- จงแสดงเซตของจุดยอดที่ใช้ระบุตำแหน่งของขอบภายในกราฟ



วิธีทำ

f(e1) = {v1 , v2 } หรือ {v2 , v1 }

f(e2) = _____

f(e3) = _____

f(e4) = _____

f(e5) = _____

f(e6) = _____

f(e7) = _____

f(e8) = _____

f(e9) = _____

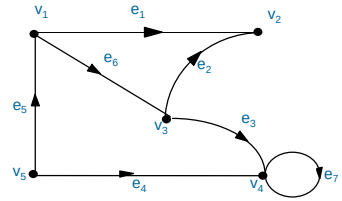
ตัวอย่าง 2

- กราฟ G = (V,E) คือ
  V= {a , b , c , d , e}
  E = { e1 , e2 , e3 , e4 , e5 , e6 , e7 , e8 , e9 }
  โดยฟังก์ชัน f : E -> V ประกอบด้วย
  f(e1) = {a,b} f(e2) = {a,c} f(e3) = {a,d}
  f(e4) = {b,c} f(e5) = {b,d} f(e6) = {b}
  f(e7) = {c,d} f(e8) = {c} f(e9) = {d}

จงแสดงภาพของกราฟตามเงื่อนไขของฟังก์ชัน f

ตัวอย่าง 3

- จากกราฟที่กำหนดให้จงแสดงส่วนประกอบของกราฟ



วิธีทำ

เมื่อ G = (V,E) จะได้ว่า

V = { _____ }

E= { _____ }

โดยฟังก์ชัน f : E -> V ประกอบด้วย

คำศัพท์ในกราฟ

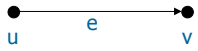
- **1)ประชิด (adjacent) และ ตกกระทบ (incident with) กรณีกราฟไม่ระบุทิศทาง** พิจารณาจากภาพ

สามารถกล่าวได้ว่า

- จุดยอด u และ v ประชิดกัน
- จุดยอด u ประชิดกับจุดยอด v
- ขอบ e ตกกระทบกับ จุดยอด u และ v
- ขอบ e เป็นขอบที่เชื่อมโยงจุดยอด u และ v
- จุดยอด u และ v เป็นจุดสิ้นสุด (endpoints) ของขอบ

คำศัพท์ในกราฟ

- **1)ประชิด (adjacent) และ ตกกระทบ (incident with) กรณีกราฟระบุทิศทาง** พิจารณาจากภาพ



สามารถกล่าวได้ว่า

- u เป็นจุดยอดเริ่มต้น (initial point)
- v เป็นจุดยอดปลายทาง (end point)
- จุดยอด u ประชิดไปยังจุดยอด v
- ขอบ e ตกกระทบจากจุดยอด u ไปยัง v
- ขอบ e เป็นขอบที่เชื่อมโยงจุดยอด u ไปยัง v
- จุดยอด u เป็นจุดเริ่มต้น (initial point) และ v เป็นจุดสิ้นสุด (end point) ของขอบ e

คำศัพท์ในกราฟ

- **2) ดีกรี (Degree) ของจุดยอด**

กรณีกราฟไม่ระบุทิศทาง ดีกรีคือจำนวนของขอบที่ตกกระทบกับจุดยอดนั้น ยกเว้น Loop ของจุดยอด 1 จุดยอดจะมี 2 ดีกรี เขียนแทนด้วย deg(v)

หมายเหตุ

- จุดยอดที่มีดีกรีเท่ากับ 0 จะเรียกว่า **isolated** นั่นคือ จุดยอดนี้ขอบด้านประชิดเลย
- จุดยอดที่มีดีกรีเท่ากับ 1 จะเรียกว่า **pendent** นั่นคือ จุดยอดนี้มีขอบประชิดเพียงด้านเดียว

คำศัพท์ในกราฟ

- **2) ดีกรี (Degree) ของจุดยอด**

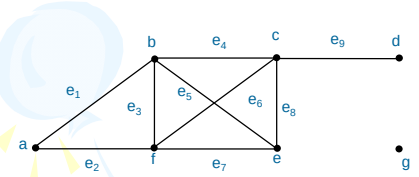
กรณีกราฟระบุทิศทาง มีการนับดีกรี 2 แบบคือ

- ดีกรีเข้า (in-degree) คือจำนวนขอบที่จุดยอดนี้เป็นจุดยอดปลายทาง เขียนแทนด้วย $\deg^-(v)$
- ดีกรีออก (out-degree) คือจำนวนขอบที่จุดยอดเป็นจุดยอดเริ่มต้น เขียนแทนด้วย $\deg^+(v)$

<u>หมายเหตุ</u> loop ในจุดยอดจะมีทั้ง ดีกรีเข้าและดีกรีออก

ตัวอย่างที่ 4

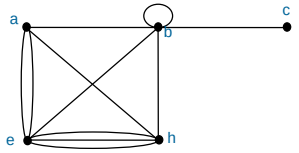
- จากกราฟด้านล่างจงตอบคำถามต่อไปนี้



จุดยอด a ประชิดกับจุดยอดใดบ้าง?
ตอบ _____
จุดยอด c ประชิดกับจุดยอดใดบ้าง?
ตอบ _____
จุดยอด e ประชิดกับจุดยอดใดบ้าง?
ตอบ _____
จุดยอด g ประชิดกับจุดยอดใดบ้าง?
ตอบ _____
ขอบ e_1 เป็นขอบตกกระทบจาก _____
ขอบ e_5 _____
ขอบ e_9 _____
ขอบ e_9 _____
deg(a) = _____
deg(f) = _____
deg(d) = _____
deg(g) = _____

ตัวอย่างที่ 5

- จากกราฟด้านล่างจงตอบคำถามต่อไปนี้
  จากภาพดีกรีของแต่ละจุดยอดสามารถแสดงได้ดังนี้
  deg(a) = _____ deg(b) = _____
  deg(c) = _____ deg(e) = _____
  deg(h) = _____



- จากลักษณะของการเชื่อมโยงของขอบ ที่ขอบ 1 ขอบ จะทำให้ 2 จุดยอดประชิดกัน เป็นผลให้นับจำนวนดีกรีของกราฟรวมได้ 2 ดีกรี (1 ขอบ = 2 ดีกรี) ดังนั้นสรุปได้ว่า ในกราฟจำนวนขอบจะเป็นครึ่งหนึ่งของจำนวนดีกรีทั้งหมดในกราฟ

ตัวอย่างที่ 5

- จากลักษณะของการเชื่อมโยงของขอบ ที่ขอบ 1 ขอบ จะทำให้ 2 จุดยอดประชิดกัน เป็นผลให้นับจำนวนดีกรีของกราฟรวมได้ 2 ดีกรี (1 ขอบ = 2 ดีกรี)
- ดังนั้นสรุปได้ว่า ในกราฟจำนวนขอบจะเป็นครึ่งหนึ่งของจำนวนดีกรีทั้งหมดในกราฟ ดังสูตรต่อไปนี้

$$2e = \sum_{v \in V} \deg(v)$$

(หมายเหตุ สูตรนี้สามารถใช้ได้กับ multiple edges และ loops)

ตัวอย่างที่ 6

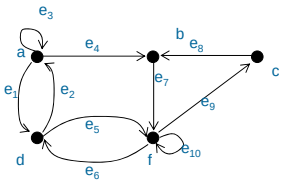
- จงหาจำนวนขอบทั้งหมด เมื่อมีจำนวนจุดยอด 10 จุดยอด และแต่ละจุดยอดมี 6 ดีกรี

วิธีทำ

ผลรวมของดีกรีที่มีทั้งหมดในกราฟคือ 6x10 = 60
จากสูตร จะได้ว่า 2e = 60
ดังนั้น จำนวนขอบในกราฟจึงมี 30 ด้าน

ตัวอย่างที่ 7

- จงตอบคำถามต่อไปนี้



a) จุดยอด a ประชิดไปยัง _____ และ _____
b) จุดยอด d ประชิดไปยัง _____ และ _____
c) จุดยอด a ประชิดไปยัง _____
d) ขอบ e_1 _____
e) ขอบ e_7 _____
f) ขอบ e_6 _____
g) ขอบ e_{10} _____
h) $\deg^-(a)$ = _____ $\deg^+(a)$ = _____
i) $\deg^-(b)$ = _____ $\deg^+(b)$ = _____
j) $\deg^-(c)$ = _____ $\deg^+(c)$ = _____
k) $\deg^-(d)$ = _____ $\deg^+(d)$ = _____
l) $\deg^-(f)$ = _____ $\deg^+(f)$ = _____

ตัวอย่างที่ 7

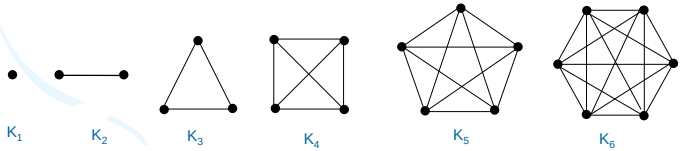
- จากตัวอย่างที่ 7 สังเกตได้ว่า ขอบ 1 ขอบ จะประกอบด้วยจุดยอดเริ่มต้นและจุดยอดสิ้นสุดอย่างละจุดยอด ดังนั้นผลรวมของดีกรีเข้าและผลรวมของดีกรีออกของทุกจุดยอดในกราฟที่มีทิศทางนั้นจะได้ผลดังนี้

$$\sum_{v \in V} \deg^{-}(v) = \sum_{v \in V} \deg^{+}(v) = |E|$$

- ซึ่งจากตัวอย่างที่ 7 เมื่อรวมจำนวนดีกรีเข้าและดีกรีออก จะมีค่าเท่ากับ _____ และมีค่าเท่ากับของขอบทั้งหมด คือ _____

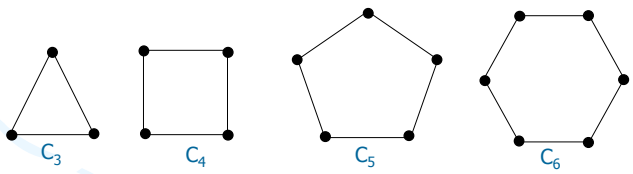
Simple graph แบบพิเศษ

- **Complete Graph** คือกราฟที่มีจำนวน n จุดยอด เขียนแทนด้วย Kn
- ซึ่งด้านทุกด้านในกราฟจะเชื่อมต่อจุดยอดกับทุกๆ จุดยอดในกราฟเข้าด้วยกัน โดยจุดยอดทุกจุดยอดสามารถที่มีด้านต่อถึงทุกจุดยอดในกราฟ



Simple graph แบบพิเศษ

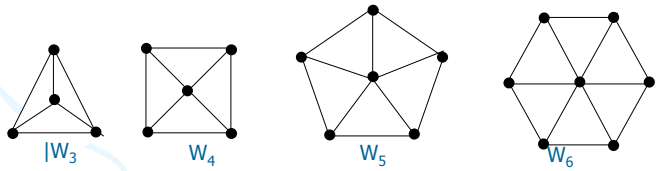
- **Cycles graph** เขียนแทนด้วยน Cn โดยที่ n เริ่มตั้งแต่ 3 ขึ้นไป
- ถ้าจุดยอดเป็น v1,v2,...,vn และ ด้านที่ได้จะเป็น {v1,v2},{v2,v3},...,{vn-1,vn} และ {vn,v1}



Simple graph แบบพิเศษ

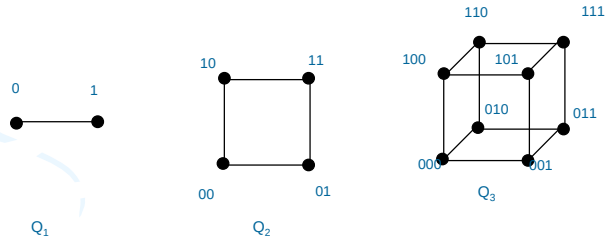
- **Wheels graph** เขียนแทนด้วย Wn

Wheel graph เกิดจากการเพิ่มจุดยอดใน Cycle graph และเชื่อมโยงจุดยอดที่เพิ่มเข้าไปนี้ไปยังจุดยอดแต่ละจุดใน Cn



Simple graph แบบพิเศษ

- **n-Cubes graph** เขียนแทนด้วย Qn เป็นกราฟที่มีใช้จุดยอดแทน bit sting ดังนั้นจำนวนของจุดยอดเท่ากับ 2n จุดยอด โดยที่ n คือความยาวของบิต bit sting โดยที่ด้านประชิดจะใช้แทนบิตที่ต่างกัน



Simple graph แบบพิเศษ

- **n-Cubes graph** เขียนแทนด้วย Qn เป็นกราฟที่มีใช้จุดยอดแทน bit sting ดังนั้นจำนวนของจุดยอดเท่ากับ 2n จุดยอด โดยที่ n คือความยาวของบิต bit sting โดยที่ด้านประชิดจะใช้แทนบิตที่ต่างกัน