Graph: Part 1 - Definition

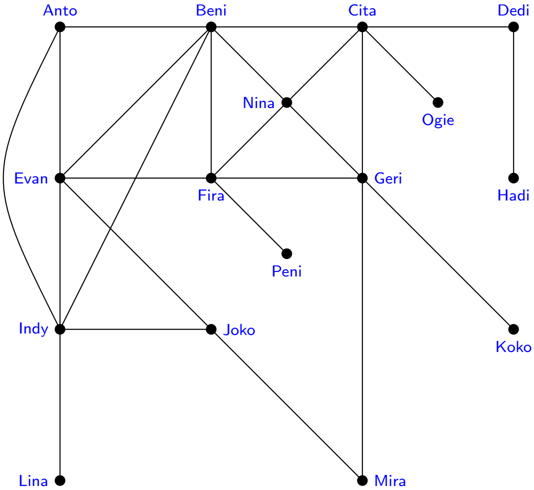
Adila A. Krisnadhi

Faculty of Computer Science, Universitas Indonesia

References and acknowledgements

- Materials of these slides are taken from:
  - Kenneth H. Rosen. *Discrete Mathematics and Its Applications*, 8ed. McGraw-Hill, 2019. Section 10.1, 10.2.
  - Jean Gallier. *Discrete Mathematics Second Edition in Progress*, 2017 [Draft]. Section 4.1, 4.2, 4.4
  - Robin J. Wilson. *Introductio to Graph Theory*, 4ed, 1996. Chapter 1 and 2.

Exhibit 1: Social Networks



Left figure: Friendship graph (modified from Rosen, Fig. 6, p.677).

Other examples: influence graphs, collaboration graphs.

Try play around with these:

- `https://mathscinet.ams.org/mathscinet/collaborationDistance.html`
- `https://www.csauthors.net/distance`

Exhibit 2: Software design

S_1 `x := 3`

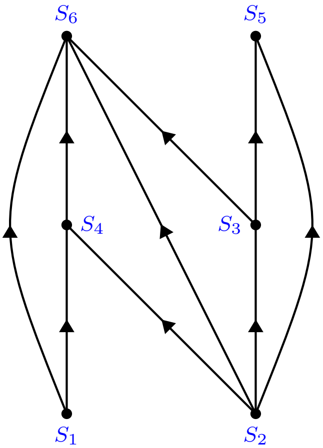
S_2 `y := 5`

S_3 `z := y + 2`

S_4 `w := y + x`

S_5 `u := z - 3`

S_6 `u := w + z`

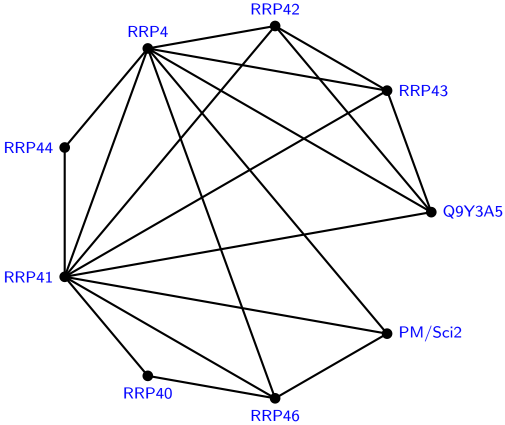


Left figure: Precedence graph (modified from Rosen, Fig. 10, p.679).

Other examples:

- module dependency graph
- function call graph
- data flow graph

Exhibit 3: Biological networks



Left: Protein interaction graph (modified from Rosen, Fig. 12, p.681)

Other examples:

- niche (food resource) overlap graph
- genetic ancestry graph
- food chain graph

Directed graph

FACULTY OF COMPUTER SCIENCE
UNIVERSITAS INDONESIA

Definition

A **directed graph (digraph)** is a tuple $G = (V, E, s, t)$ where

- V is a nonempty set of **nodes**/**vertices**;
- E is a possibly empty set of **edges**/**arcs**;
- $s\colon E \to V$, called the **source function**, maps each edge $e \in E$ to a source node $s(e)$ of e;
- $t\colon E \to V$, called the **target function**, maps each edge $e \in E$ to a target node $t(e)$ of e.

The source and target of an edge are called its **endpoints**. Also, we say that each edge **connects** both its endpoints or **connects** its source to its target.

If we don't care about the source and target functions, we simply write $G = (V, E)$.

Undirected graph

Undirected graphs are just directed graphs whose edges point to both directions. That is, each edge is associated to a set of nodes $\{u, v\}$ with $u = v$ allowed.

Definition

An **(undirected) graph** is a tuple $G = (V, E, st)$ where

- V is a nonempty set of **nodes**/**vertices**;
- E is a possibly empty set of **edges**/**arcs**;
- $st\colon E \to V \cup [V]^2$ is a function that maps an edge e to a set of its endpoint node(s) where $[V]^2 = \{\{u, v\} \in 2^V \mid u \neq v\}$.

If we don't care about the function st, we simply write a graph as a pair $G = (V, E)$.

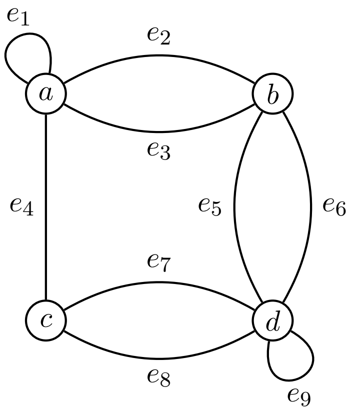
Basic terminology (1)

- A graph is called an **infinite graph** if either its set of nodes or its set of edges is infinite. Otherwise, the graph is **finite**.
- If V is the set of nodes of graph G, the **order** of G is $|V|$, i.e., the cardinality of V.
- In a directed graph, each edge has a direction: from its source to its target. Thus, we call such an edge a **directed edge**, drawn as a unidirectional arrow. .
- In an undirected graph, an edge does not have a source and a target. Rather, it simply has two endpoints, which may even be the same node. Hence, such an edge is sometimes called **undirected edge**, drawn as a simple line segment (no arrow tip) connecting both its endpoints.

Basic terminology (2)

- Loops:
  - A directed edge e is a **loop** if $s(e) = t(e)$.
  - An undirected edge e is a **loop** if $st(e) = \{u\}$ for a single node u.
- Parallel edges:
  - Two directed edges e_1, e_2 are **parallel** if $s(e_1) = s(e_2)$ and $t(e_1) = t(e_2)$
  - Two undirected edges e_1, e_2 are **parallel** if $st(e_1) = st(e_2)$.
- A (directed/undirected) graph is **simple** iff it has no parallel edges and no loop.
- A (directed/undirected) **multigraph** is a graph in which parallel edges are allowed.
  - Every simple graph is thus a multigraph without parallel edges.

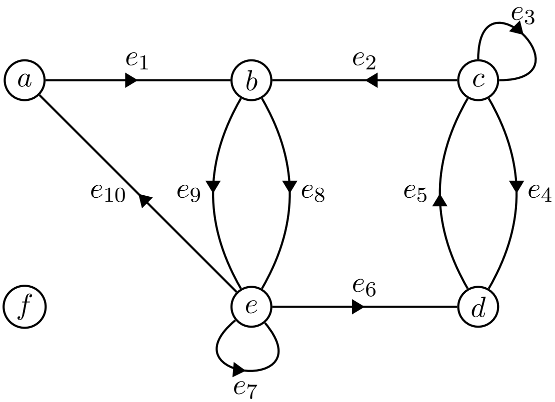
Example



Graph $G = (V, E, st)$ is undirected where

- $V = \{a, b, c, d\}$
- $E = \{e_1, \ldots, e_9\}$
- $st(e_1) = \{a\}$, $st(e_2) = st(e_3) = \{a, b\}$, $st(e_4) = \{a, c\}$, $st(e_5) = st(e_6) = \{b, d\}$, $st(e_7) = st(e_8) = \{c, d\}$, $st(e_9) = \{d\}$.
- Parallel edges: e_2 and e_3, e_5 and e_6, e_7 and e_8
- Loops: e_1, e_9

Example



Graph $G = (V, E, s, t)$ is directed:

- $V = \{a, b, c, d, e, f\}$
- $E = \{e_1, \ldots, e_{10}\}$
- $s(e_1) = a$,
  $s(e_2) = s(e_3) = s(e_4) = c$,
  $s(e_5) = d$,
  $s(e_6) = s(e_7) = s(e_{10}) = e$,
  $s(e_8) = s(e_9) = b$
- $t(e_1) = t(e_2) = b$,
  $t(e_3) = t(e_5) = c$,
  $t(e_4) = t(e_6) = d$,
  $t(e_7) = t(e_8) = t(e_9) = e$,
  $t(e_{10}) = a$
- Parallel edges: e_8 and e_9.
- Loops: e_3, e_7.

Exercise

1. Define a graph G whose nodes are Indonesian provinces in the island of Sumatra such that two nodes are connected by an edge iff the two provinces share a land border. Is the graph directed or undirected? Is it simple?
2. Define a graph G whose nodes are all bit strings of length 2 such that bit string a is connected to bit string b iff b can be obtained from a by concatenating a single bit (either 0 or 1) to the rightmost position of a and deleting its leftmost bit. For example, 00 is connected to 01 because we can obtain 01 from 00 as follows: $00 \rightarrow 001 \rightarrow 01$. Is the graph directed or undirected? Is it simple?

Define a graph G whose nodes are Indonesian provinces in the island of Sumatra such that two nodes are connected by an edge iff the two provinces share a land border.

Define a graph G whose nodes are all bit strings of length 2 such that bit string a is connected to bit string b iff b can be obtained from a by concatenating a single bit (either 0 or 1) to the rightmost position of a and deleting its leftmost bit.

Adjacency and incidence

- Let u, v be two (possibly the same) nodes and e an edge in a digraph $G = (V, E, s, t)$ such that $s(e) = u$ and $t(e) = v$. Then, we say that:
  - u is **connected to** v and v is **connected from** u by the edge e (we sometimes write e as the pair (u, v) if e is not parallel to another edge);
  - v is **adjacent to** u (Note: adjacency in digraph is not symmetric).
- Let u, v be two (possibly the same) nodes and e an edge in an undirected graph $G = (V, E, st)$ such that $st(e) = \{u, v\}$. Then, we say that:
  - u and v are **connected** by e (we sometimes write e as the set $\{u, v\}$ if e is not parallel to another edge)
  - u and v are **adjacent** (Adjacency is symmetric in undirected graphs).
- If an edge e connects u and v, then we also say that e is **incident** to/at both u and v, and the nodes u and v are **incident** to/with e.

Neighborhood

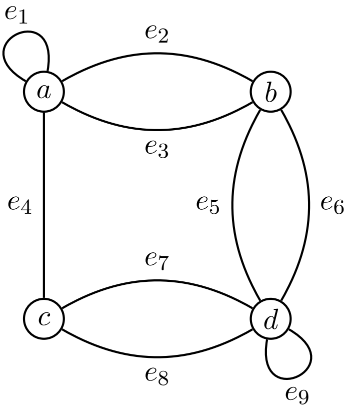
Let u be a node in a (directed/undirected graph) $G = (V, E)$.

- The **neighborhood** of u is the set $N_G(u) = \{v \in V \mid v \text{ is adjacent to } u\}$
- The **proper neighborhood** of u is the set $NP_G(u) = \{v \in V \mid v \text{ is adjacent to } u, v \neq u\}$

The above notions can be extended to a set of nodes. Given a graph $G = (V, E)$, let $U \subseteq V$ be a subset of nodes of G.

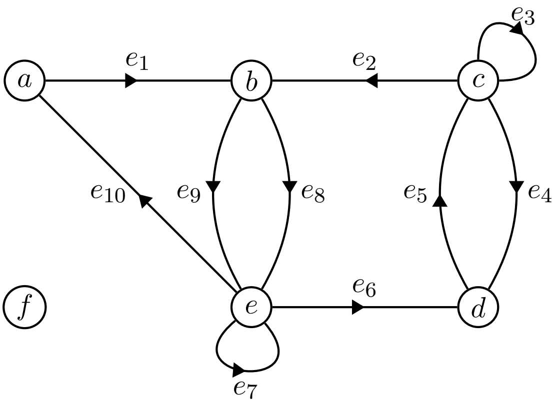
- The **neighborhood** of U is the set $N_G(U) = \{v \in V \mid v \text{ is adjacent to } u \text{ for some } u \in U\}$.
  - Note that $N_G(U) = \bigcup_{u \in U} N_G(u)$.
- The **proper neighborhood** of U is the set $NP_G(U) = \{v \in V - U \mid v \text{ is adjacent to } u \text{ for some } u \in U\}$

Determine the neighborhood and proper neighborhood of a and $\{b, d\}$.



Example

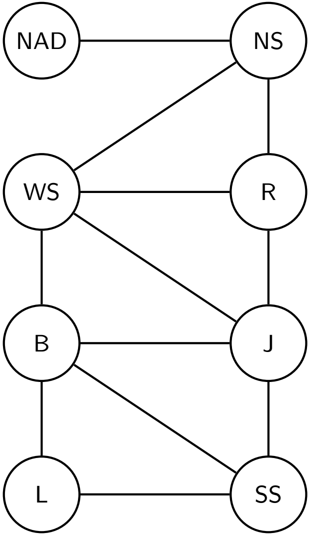
Determine the neighborhood and proper neighborhood of a, c, and $\{b, d\}$.



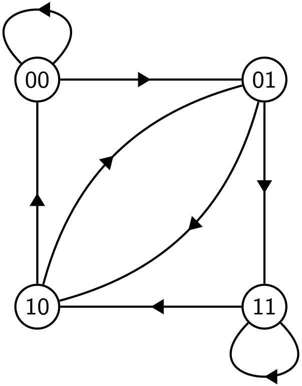
Determine each node's neighborhood in the graphs below (defined for exercises in Slide 12):

1. Graph G whose nodes are Indonesian provinces in the island of Sumatra such that two nodes are connected by an edge iff the two provinces share a land border.
2. Graph G whose nodes are all bit strings of length 2 such that bit string a is connected to bit string b iff b can be obtained from a by concatenating a single bit (either 0 or 1) to the rightmost position of a and deleting its leftmost bit.

Graph G: nodes: Indonesian provinces in Sumatra; edges: provinces sharing land border. Find each node's neighborhood.



Graph G: nodes: bit strings of length 2; edges: e connects u to v if v is obtained from u by concatenating a single bit to its rightmost position and deleting its leftmost bit. Find each node's neighborhood.



Degree of nodes in an undirected graph

Definition

Degree of a node v, denoted $\deg(v)$, in an undirected graph $G = (V, E, s, t)$ is:

$$\deg(v) = |\{e \in E \mid v \in st(e)\}| + |\{e \in E \mid st(e) = \{v\}\}|$$

That is, degree of v is the number of edges incident with v, except that each loop at v contributes twice to the degree of v.

- If $\deg(v) = 0$, we call v **isolated**.
- If $\deg(v) = 1$, we call v a **pendant**.

Degree of nodes in a digraph

Definition

Let $G = (V, E, s, t)$ be a digraph.

- **In-degree** of a node v, denoted $\deg^-(v)$, in G is:

$$\deg^-(v) = |\{e \in E \mid v = t(e)\}|$$

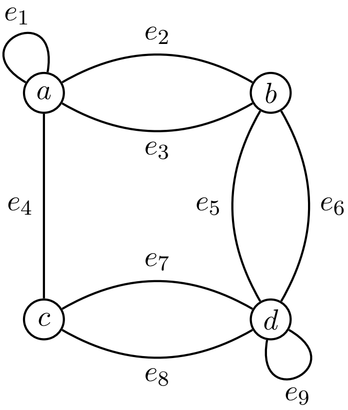
  That is, in-degree of v is the number of incoming edges to v.
- **Out-degree** of a node v, denoted $\deg^+(v)$, in G is:

$$\deg^+(v) = |\{e \in E \mid v = s(e)\}|$$

  That is, out-degree of v is the number of outgoing edges from v.
- **Degree** of a node v, denoted $\deg(v)$, in G is $\deg(v) = \deg^+(v) + \deg^-(v)$.

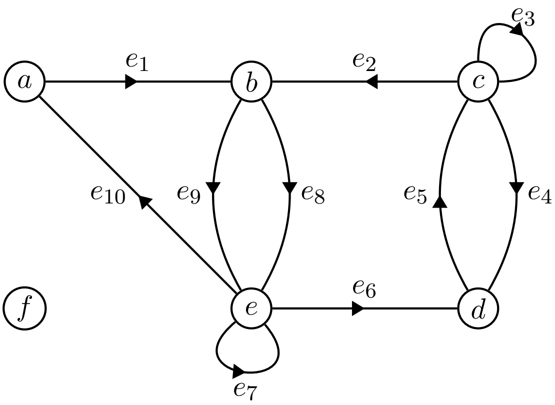
Example

Determine the degree of each node in the graph below.



Example

Determine the degree of each node in the graph below.

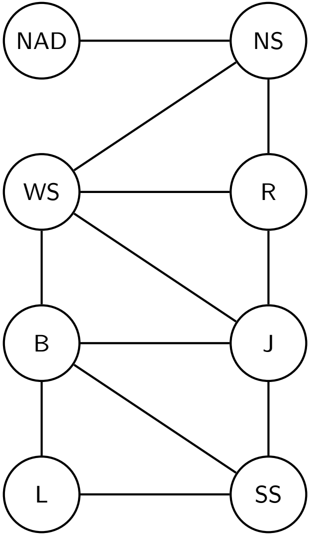


Exercise

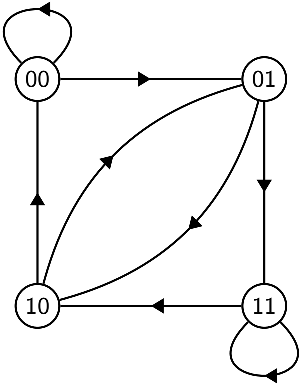
Determine the degree of each node in the graphs below (defined for exercises in Slide 12):

1. Graph G whose nodes are Indonesian provinces in the island of Sumatra such that two nodes are connected by an edge iff the two provinces share a land border.
2. Graph G whose nodes are all bit strings of length 2 such that bit string a is connected to bit string b iff b can be obtained from a by concatenating a single bit (either 0 or 1) to the rightmost position of a and deleting its leftmost bit.

Graph G: nodes: Indonesian provinces in Sumatra; edges: provinces sharing land border. Find the degree of each of its nodes.



Graph G: nodes: bit strings of length 2; edges: e connects u to v if v is obtained from u by concatenating a single bit to its rightmost position and deleting its leftmost bit. Find the degree of each of its nodes.



Handshaking theorem

Theorem (Handshaking theorem)

Let $G = (V, E)$ be a (directed/undirected) graph and $|E|$ is the number of its edges.
Then, $2|E| = \sum_{v \in V} \deg(v)$.
Moreover, if G is a digraph, then $|E| = \sum_{v \in V} \deg^{+}(v) = \sum_{v \in V} \deg^{-}(v)$.

Corollary of the handshaking theorem

Theorem

Let $G = (V, E)$ be a (directed/undirected) graph. Then, there are an even number of nodes whose degree is odd.

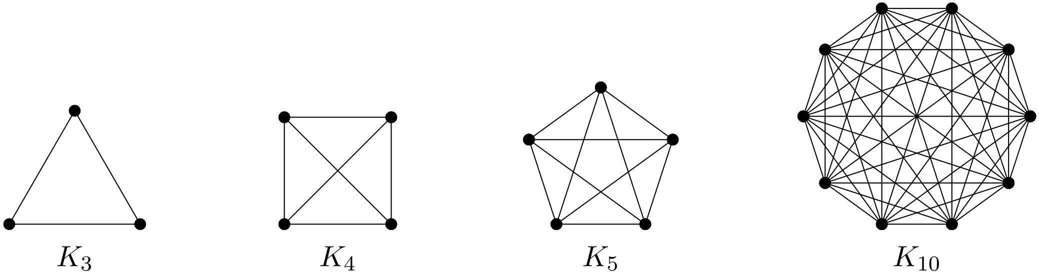
Exercise

Prove that in a party attended by an odd number of people, there exists a person who is acquainted with an even number of others. (This includes being acquainted with no one, i.e., with zero other people).

Special types of undirected graphs: Null and complete graphs

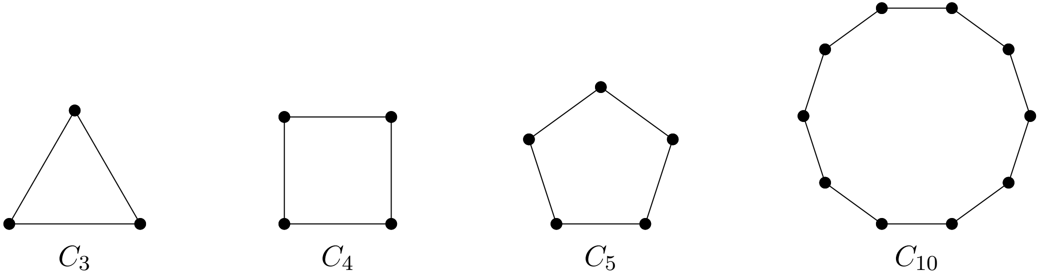
Null graph: graph with n nodes but no edge.

Complete graph K_n: with n nodes: simple graph where every pair of nodes are adjacent.



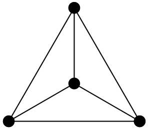
Special types of undirected graphs: Cycle graphs

Cycle graph C_n with n nodes $\{u_1, \ldots, u_n\}$ has exactly one edge between u_i and u_{i+1} as well as between u_n and u_1. Cycle graphs are useful, e.g., for modeling topology of a local area network (LAN)

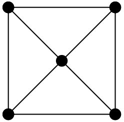


Special types of undirected graphs: Wheel graph

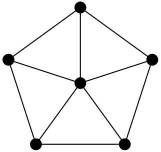
Wheel graph W_n has $n+1$ nodes and is obtained from a cycle graph C_n by adding one node that is connected (with a single edge) to every other nodes. Like cycle graphs, wheel graphs are also useful for modeling the topology of LAN.



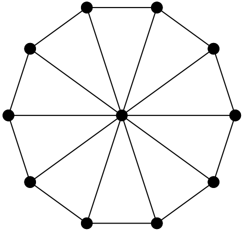
W_3



W_4

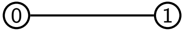


W_5

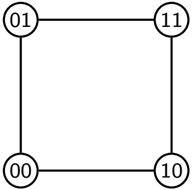


W_{10}

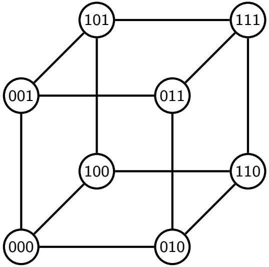
n-**dimensional hypercube** or n-**cube**, denoted Q_n, is a simple graph with 2^n nodes, each representing a bit string of length n. Node a and b are adjacent if the bit strings representing a and b differ in exactly one bit position.



Q_1



Q_2



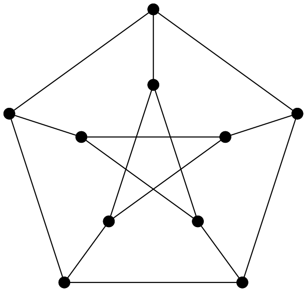
Q_3

Regular graphs

Regular graph of degree k is a graph whose all of its nodes have the same degree k.

- Examples: Null graphs (degree 0), complete graphs K_n (degree $n-1$), cycle graphs (degree 2), hypercubes Q_n (degree n), Petersen graph (degree 3), Platonic graphs (degrees 3, 4, 5, etc.).
- Regular graphs may include loops or parallel edges.
- This definition is also applicable to digraphs.

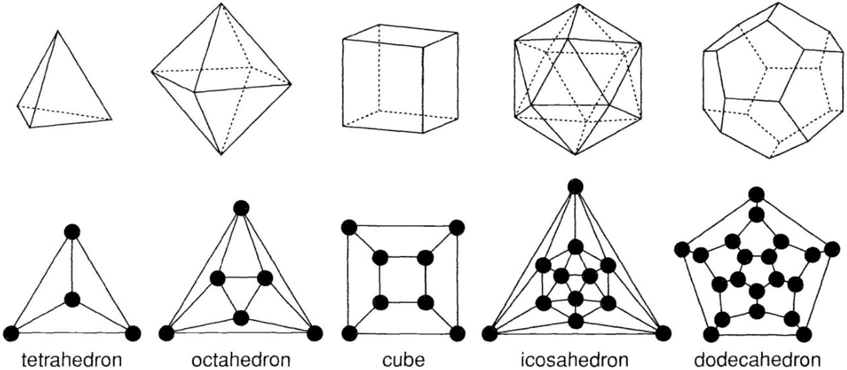
Petersen graph



Petersen graph has many interesting properties (see Wikipedia page on Petersen graph):

- Nonplanar
- Smallest hypohamiltonian graph: has a Hamiltonian path, but not Hamiltonian cycle, and deleting any one node causes it to have a Hamiltonian cycle.
- Coloring the nodes so that no two adjacent nodes have the same color requires at least 3 colors.
- etc.

Platonic graphs



Wilson, Fig. 3.5, p.18