Lecture 5: Trees (3) and Circuits

Shuai Li

John Hopcroft Center, Shanghai Jiao Tong University

https://shuaili8.github.io

https://shuaili8.github.io/Teaching/CS445/index.html

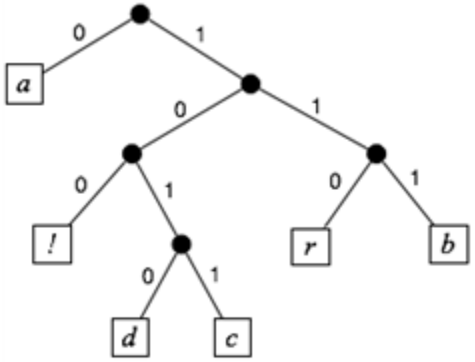
Wiener index

- In a communication network, large diameter may be acceptable if most pairs can communicate via short paths. This leads us to study the average distance instead of the maximum
- Wiener index $D(G) = \sum_{u,v \in V(G)} d_G(u, v)$
- Theorem (2.1.14, W) Among trees with n vertices, the Wiener index $D(T)$ is minimized by stars and maximized by paths, both uniquely

Prefix coding

- A binary tree is a rooted plane tree where each vertex has at most two children
- Given large computer files and limited storage, we want to encode characters as binary lists to minimize total length
- Prefix coding: no code word is an initial portion of another

- Example: 11001111011



character	encoding
a	0
b	111
c	1011
d	1010
r	110
$!$	100

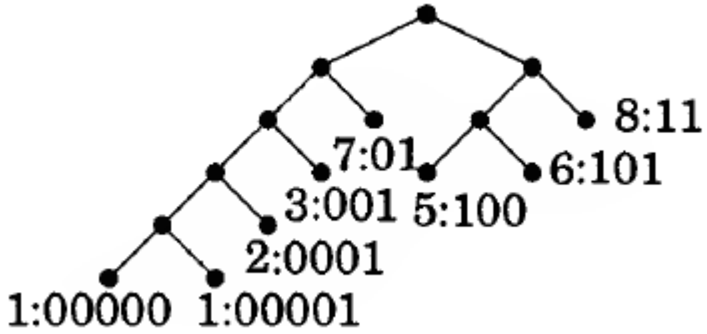
A binary prefix code for the alphabet $\{a, b, c, d, r, !\}$

Huffman coding

- Input: Weights (frequencies or probabilities) $p_1, \ldots, p_n$
- Output: Prefix-free code (equivalently, a binary tree)
- Idea: Infrequent items should have longer codes; put infrequent items deeper by combining them into parent nodes.
- Recursion: replace the two least likely items with probabilities p, p' with a single item of weight $p + p'$

Example (2.3.14, W)

a	5	100
b	1	00000
c	1	00001
d	7	01
e	8	11
f	2	0001
g	3	001
h	6	101



The average length is $\frac{5\times3+5+5+7\times2+\cdots}{33}=\frac{30}{11}<3$

Huffman coding is optimal

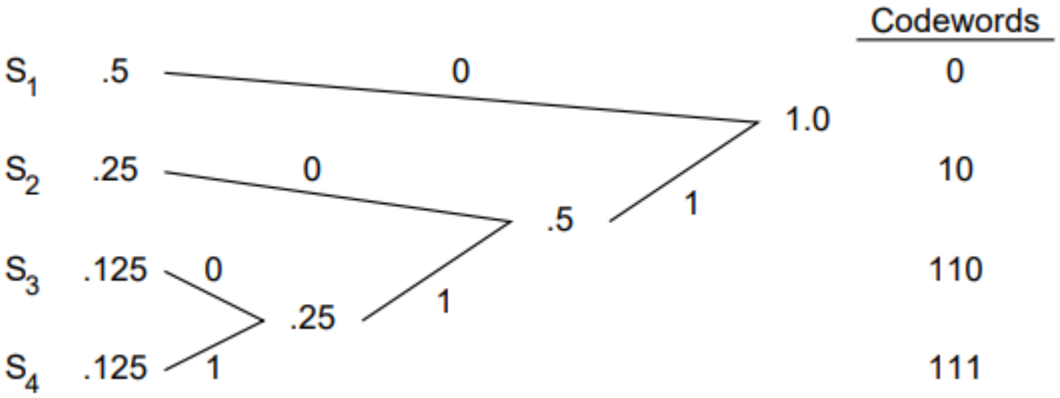
- Theorem (2.3.15, W) Given a probability distribution $\{p_i\}$ on n items, Huffman's Algorithm produces the prefix-free code with minimum expected length

Huffman coding and entropy

- The entropy of a discrete probability distribution $\{p_i\}$ is that

$$H(p) = -\sum_i p_i \log_2 p_i$$

- $H(p) \leq$ average length of Huffman coding $\leq H + 1$
- When each p_i is a power of ½, average length of Huffman coding is $H(p)$



$$\begin{aligned} \text{average length} &= (1)\left(\frac{1}{2}\right) + (2)\left(\frac{1}{4}\right) + (3)\left(\frac{1}{8}\right) + (3)\left(\frac{1}{8}\right) \\ &= 1.75 \;\; \text{bits/symbol} \end{aligned}$$

$$\begin{aligned} H &= \frac{1}{2}\log_2 2 + \frac{1}{4}\log_2 4 + \frac{1}{8}\log_2 8 + \frac{1}{8}\log_2 8 \\ &= \frac{1}{2} + \frac{1}{2} + \frac{3}{8} + \frac{3}{8} \\ &= 1.75 \end{aligned}$$

Circuits

Eulerian circuit

- A closed walk through a graph using every edge once is called an Eulerian circuit
- A graph that has such a walk is called an Eulerian graph

- Theorem (1.2.26, W) A graph G is Eulerian $\Longleftrightarrow$ it has at most one nontrivial component and its vertices all have even degree
- (possibly with multiple edges)
- Proof "$\Longrightarrow$" That G must be connected is obvious.
  Since the path enters a vertex through some edge and leaves by another edge, it is clear that all degrees must be even

Key lemma

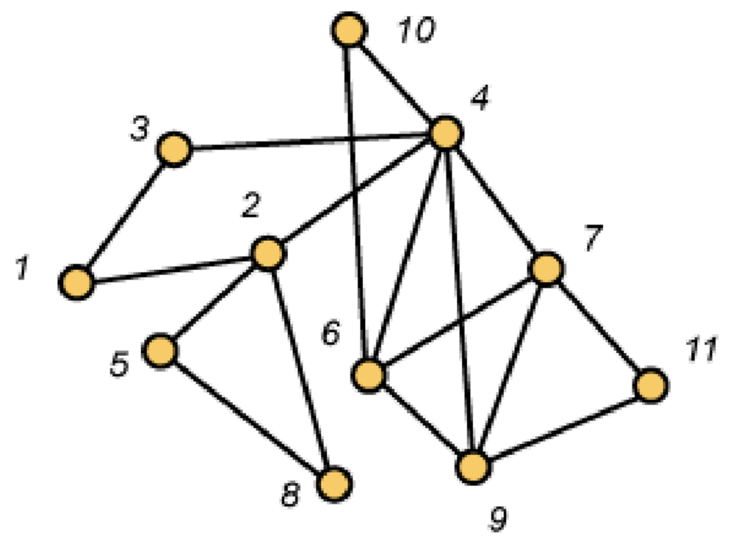
- Lemma (1.2.25, W) If every vertex of a graph G has degree at least 2, then G contains a cycle.

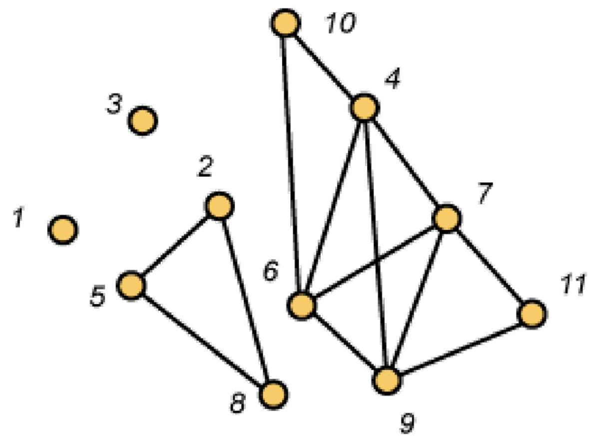
Hierholzer's Algorithm for Euler Circuits

1. Choose a root vertex r and start with the trivial partial circuit (r)
2. Given a partial circuit $(x_0, e_1, x_1, \ldots, x_{t-1}, e_t, x_t = x_0)$ that traverses not all edges of G, remove these edges from G
3. Let i be the least integer for which x_i is incident with one of the remaining edges
4. Form a greedy partial circuit among the remaining edges of the form $(x_i = y_0, e_1', y_1, \ldots, y_{s-1}, e_s', y_s = x_i)$
5. Expand the original circuit by setting $(x_0, e_1, \ldots, e_i, x_i = y_0, e_1', y_1, \ldots, y_{s-1}, e_s', y_s = x_i, e_{i+1}, \ldots, e_t, x_t = x_0)$
6. Repeat step 2-5

Example

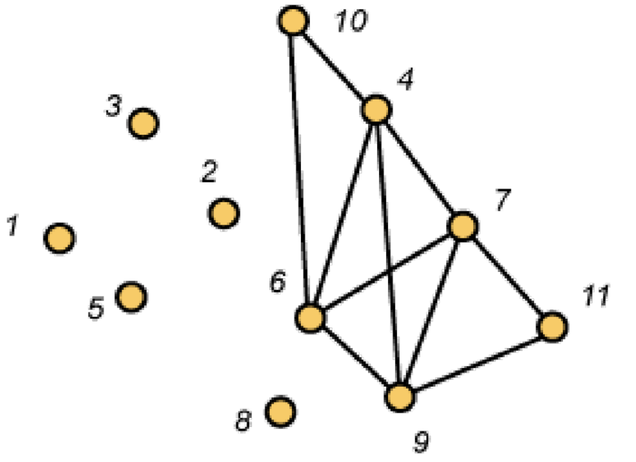
1. Start with the trivial circuit (1)
2. Greedy algorithm yields the partial circuit
   (1,2,4,3,1)
3. Remove these edges
4. The first vertex incident with remaining edges is 2
5. Greedy algorithms yields (2,5,8,2)
6. Expanding (1,2,5,8,2,4,3,1)
7. Remove these edges

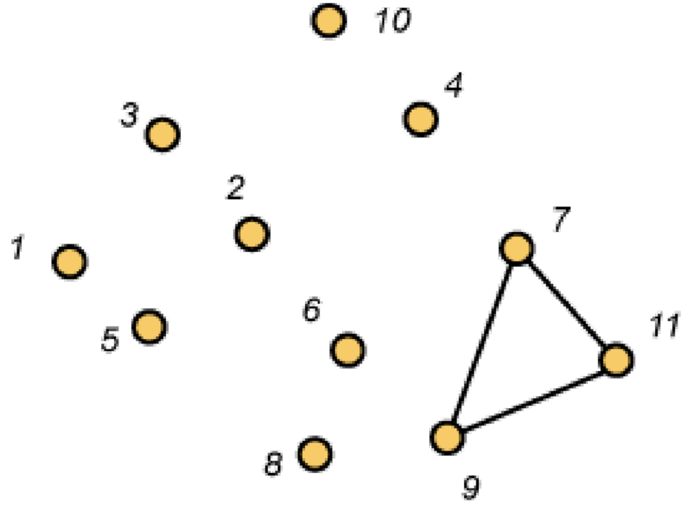




Example (cont.)

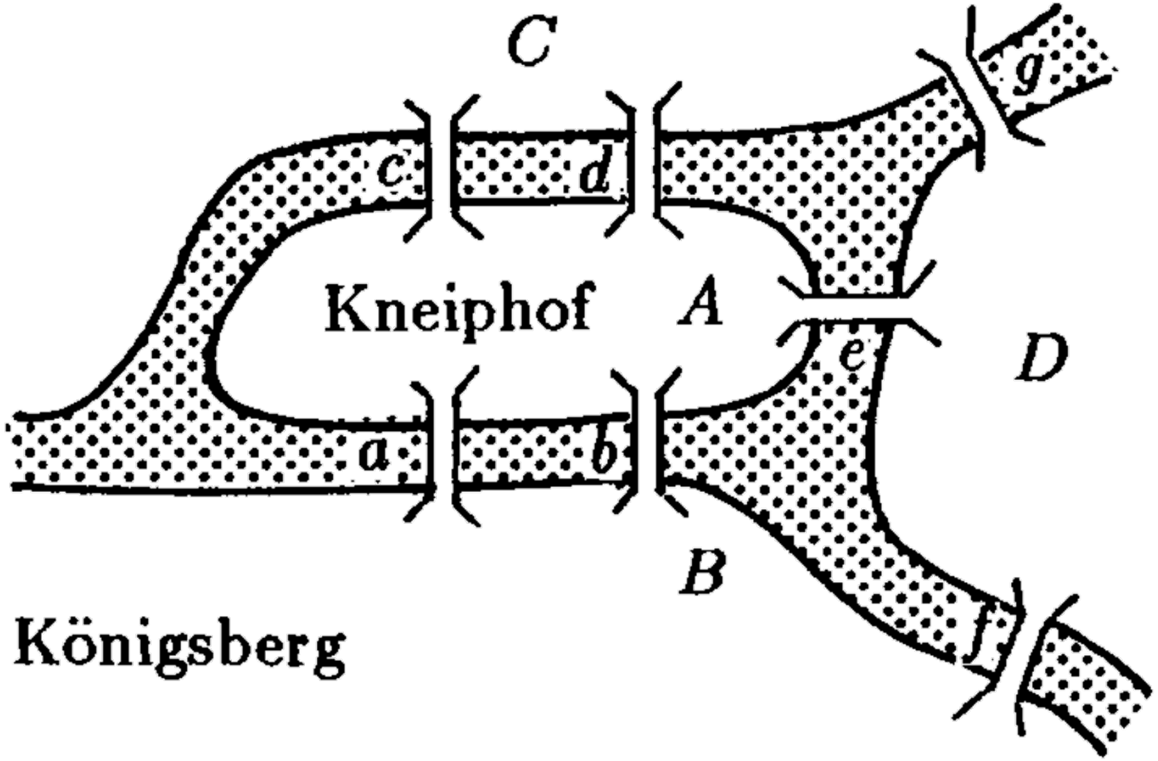
6. Expanding (1,2,5,8,2,4,3,1)
7. Remove these edges
8. First vertex incident with remaining edges is 4
9. Greedy algorithm yields (4,6,7,4,9,6,10,4)
10. Expanding (1,2,5,8,2,4,6,7,4,9,6,10,4,3,1)
11. Remove these edges
12. First vertex incident with remaining edges is 7
13. Greedy algorithm yields (7,9,11,7)
14. Expanding (1,2,5,8,2,4,6,7,9,11,7,4,9,6,10,4,3,1)

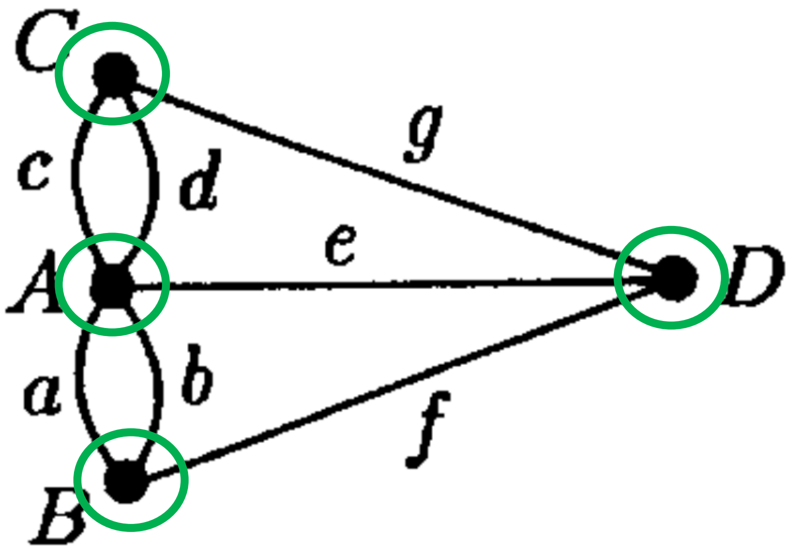




Eulerian circuit

- Theorem (1.2.26, W) A graph G is Eulerian $\Leftrightarrow$ it has at most one nontrivial component and its vertices all have even degree



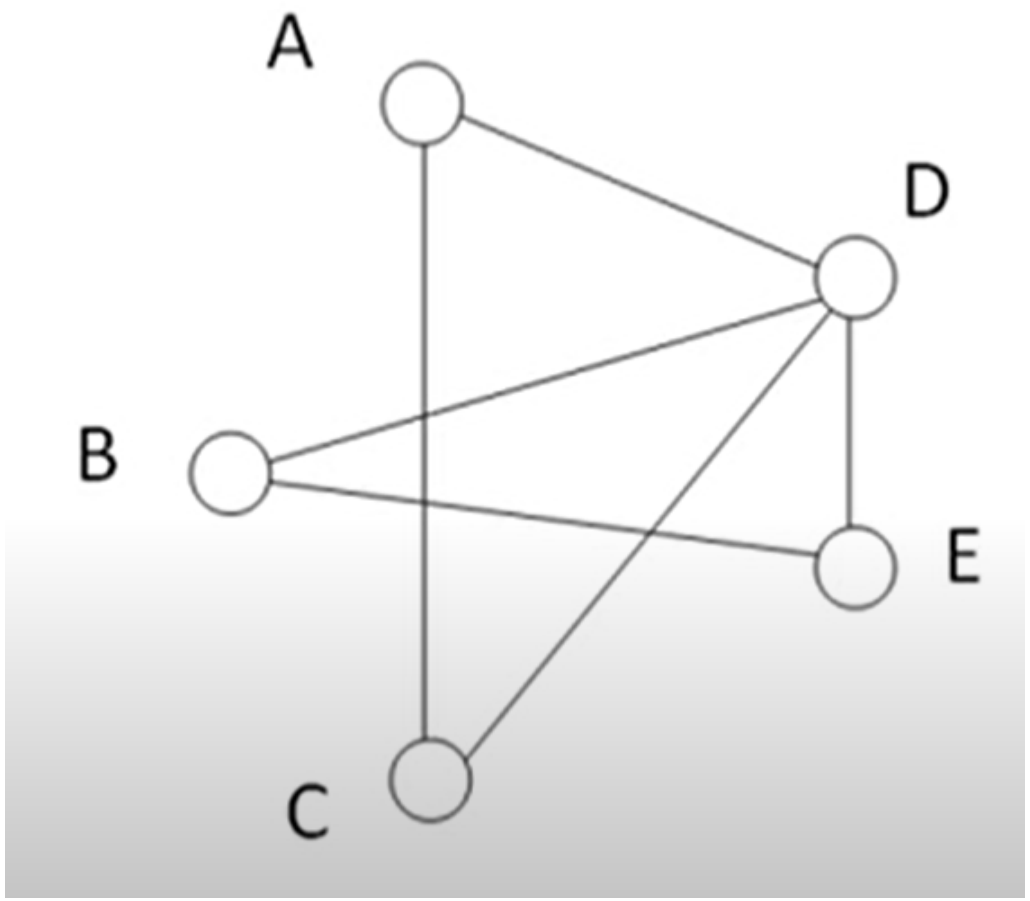


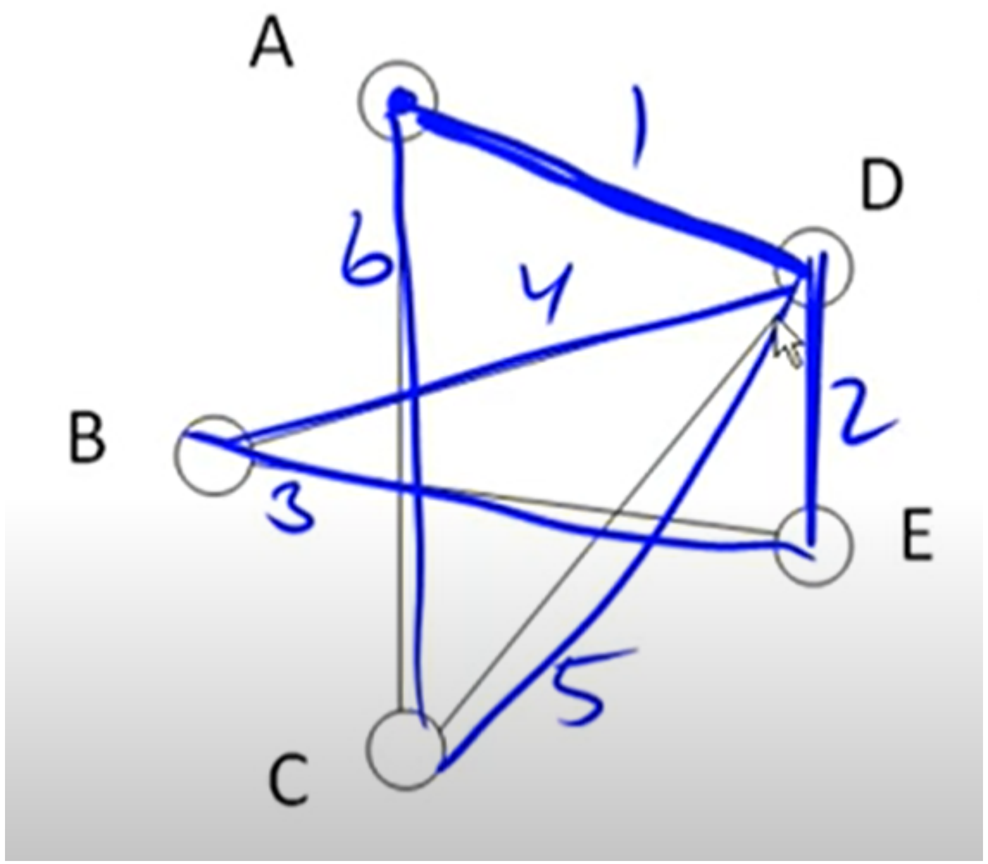
Fleury's Algorithm for Identifying Eulerian Circuits

- (Ex3, S1.4.2, H)
- Given: An Eulerian graph *G*, with all of its edges *unmarked*

1. Choose a vertex *v*, and call it the "lead vertex"
2. If all edges of *G* have been marked, then stop. Otherwise continue to step 3
3. Among all edges incident with the lead vertex, choose, if possible, one that is not a bridge of the subgraph formed by the unmarked edges. If this is not possible, choose any edge incident with the lead. Mark this edge and let its other end vertex be the new lead vertex
4. Go to step 2

Example





Other properties

- Proposition (1.2.27, W) Every even graph decomposes into cycles

- The necessary and sufficient condition for a directed Eulerian circuit is that the graph is connected and that each vertex has the same ‘in-degree’ as ‘out-degree’

TONCAS

- TONCAS: The obvious necessary condition is also sufficient
- Theorem (1.2.26, W) A graph G is Eulerian $\Longleftrightarrow$ it has at most one nontrivial component and its vertices all have even degree

- Proposition (1.3.28, W) The nonnegative integers $d_1, \dots, d_n$ are the vertex degrees of some graph $\Longleftrightarrow$ $\sum_{i=1}^{n} d_i$ is even
- (Possibly with loops)
- Otherwise $(2,0,0)$ is not realizable

1.3.63. (!) Let $d_1, \dots, d_n$ be integers such that $d_1 \geq \cdots \geq d_n \geq 0$. Prove that there is a loopless graph (multiple edges allowed) with degree sequence $d_1, \dots, d_n$ if and only if $\sum d_i$ is even and $d_1 \leq d_2 + \cdots + d_n$. (Hakimi [1962])

Hamiltonian path/circuits

- A path P is Hamiltonian if $V(P) = V(G)$
  - Any graph contains a Hamiltonian path is called traceable
- A cycle C is called Hamiltonian if it spans all vertices of G
  - A graph is called Hamiltonian if it contains a Hamiltonian circuit
- In the mid-19th century, Sir William Rowan Hamilton tried to popularize the exercise of finding such a closed path in the graph of the dodecahedron

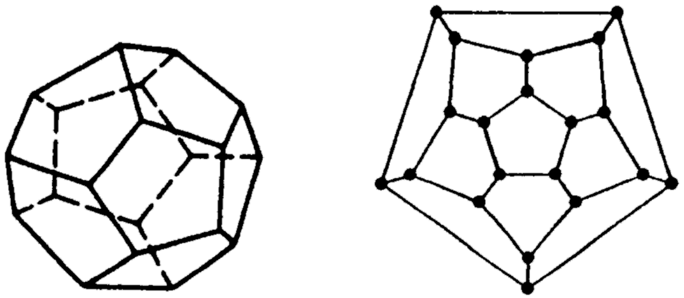
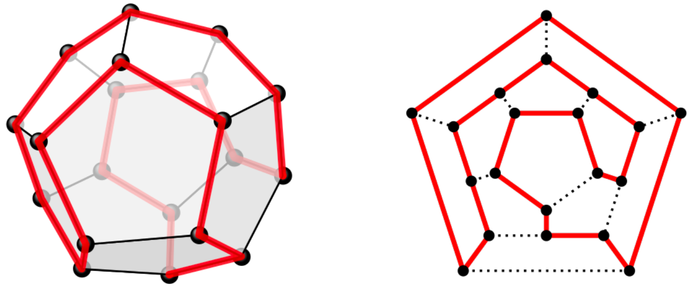


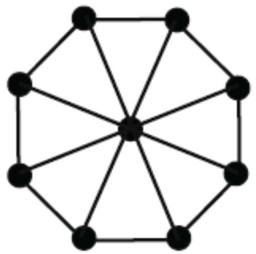
Figure 1.9



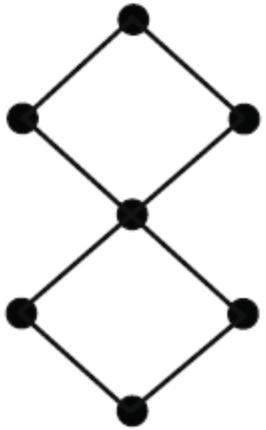
Degree parity is not a criterion

Theorem (1.2.26, W) A graph G is Eulerian $\Leftrightarrow$ it has at most one nontrivial component and its vertices all have even degree

- Hamiltonian graphs
  - all even degrees C_{10}
  - all odd degrees K_{10}
  - a mixture G_1
- non-Hamiltonian graphs
  - all even G_2
  - all odd $K_{5,7}$
  - mixed P_9



G_1



G_2

Example

- The Petersen graph has a Hamiltonian path but no Hamiltonian cycle

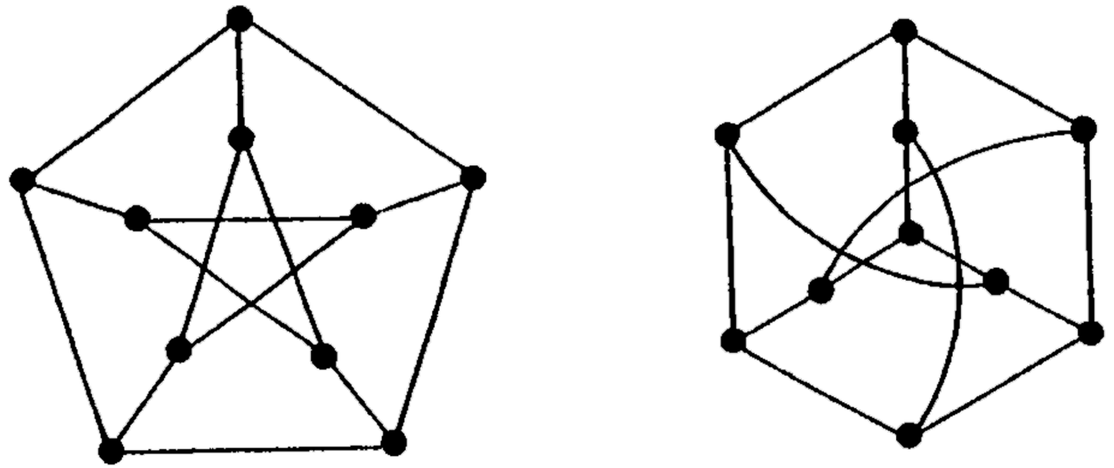
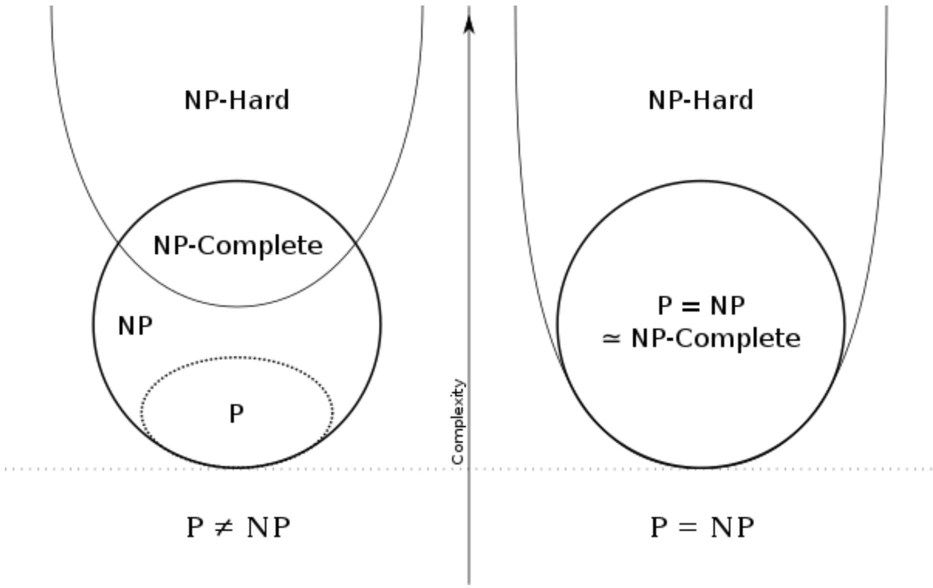


Figure 1.4

- Determining whether such paths and cycles exist in graphs is the Hamiltonian path problem, which is NP-complete.

P, NP, NPC, NP-hard

- P The general class of questions for which some algorithm can provide an answer in polynomial time
- NP The class of questions for which an answer can be *verified* in polynomial time
- NP-Complete
  1. c is in NP
  2. Every problem in NP is reducible to c in polynomial time
- NP-hard
  - ~~c is in NP~~
  - Every problem in NP is reducible to c in polynomial time



Large minimal degree implies Hamiltonian

- Theorem (1.22, H, Dirac) Let G be a graph of order $n \geq 3$. If $\delta(G) \geq n/2$, then G is Hamiltonian

> Proposition (1.3.15, W) If $\delta(G) \geq \frac{n-1}{2}$, then G is connected
>
> (Ex16, S1.1.2, H) (1.3.16, W)
> If $\delta(G) \geq \frac{n-2}{2}$, then G need not be connected

- The bound is tight
  (Ex12b, S1.4.3, H) $G = K_{r,r+1}$ is not Hamiltonian
- The condition is not necessary
  - C_n is Hamiltonian but with small minimum (and even maximum) degree

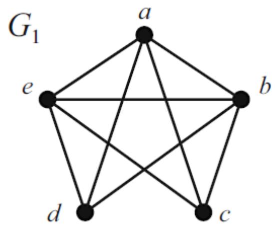
Generalized version

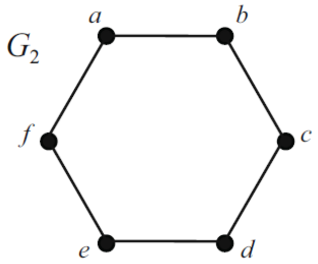
- Exercise (Theorem 1.23, H, Ore; Ex3, S1.4.3, H) Let G be a graph of order $n \geq 3$. If $\deg(x) + \deg(y) \geq n$ for all pairs of nonadjacent vertices x, y, then G is Hamiltonian

Theorem (1.22, H, Dirac) Let G be a graph of order $n \geq 3$. If $\delta(G) \geq n/2$, then G is Hamiltonian

Independence number & Hamiltonian

- A set of vertices in a graph is called independent if they are pairwise nonadjacent
- The independence number of a graph G, denoted as $\alpha(G)$, is the largest size of an independent set
- Example: $\alpha(G_1) = 2, \alpha(G_2) = 3$
- Theorem (1.24, H) Let G be a connected graph of order $n \geq 3$. If $\kappa(G) \geq \alpha(G)$, then G is Hamiltonian
- The result is tight: $\kappa(G) \geq \alpha(G) - 1$ is not enough
  - $K_{r,r+1}$: $\kappa = \mathrm{r}, \alpha = r + 1$
  - Peterson graph: $\kappa = 3, \alpha = 4$ (Ex4, S1.4.3, H)





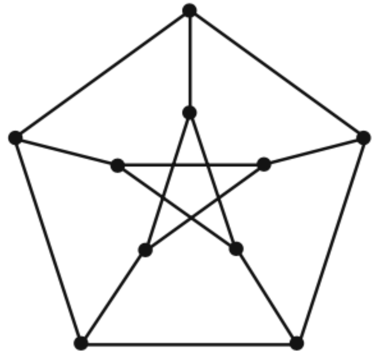
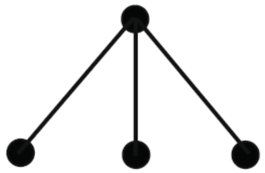
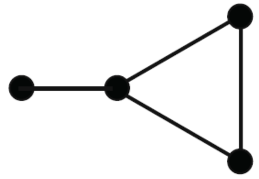


FIGURE 1.63. The Petersen Graph.

Pattern-free & Hamiltonian



$K_{1,3}$



Z_1

- G is H-free if G doesn't contain a copy of H as induced subgraph
- Theorem (1.25, H) If G is 2-connected and $\{K_{1,3}, Z_1\}$-free, then G is Hamiltonian

(Ex14, S1.1.2, H) $\kappa(G) \geq 2$ implies G has at least one cycle

- The condition 2-connectivity is necessary
- (Ex2, S1.4.3, H) If G is Hamiltonian, then G is 2-connected