Undirected Graphs

Lecture 10

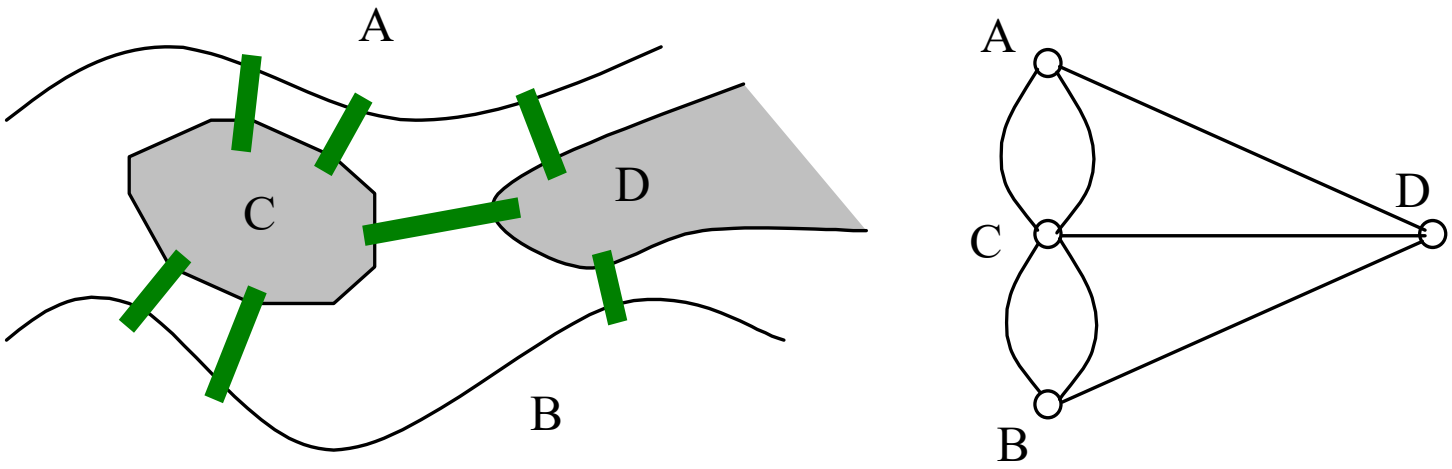
Discrete Mathematical Structures

Undirected Graphs

- Part I: Graph as a Model
  - Paths and circuits in a graph
  - Subgraph and quotient graph
  - Graph as a model for problem solving
- Part II: Traversal in a Graph
  - Euler paths and circuits
  - Hamiltonian paths and circuits

Seven Bridges at Königsberg

- Abstraction
  - Vertices representing objects - areas
  - Edges representing the relationship between objects – connected by a bridge



Graph and Diagram

- Graph G is a triple: $G = \langle V_G, E_G, \varphi \rangle$
  - V_G and E_G are sets， satisfying $V_G \cap E_G = \phi$, $\varphi: E_G \rightarrow \{\{v_i, v_j\} | v_i, v_j \in V_G\}$
    Note: $\{v_i, v_j\} = \{v_j, v_i\}$
  - A graph can be represented conveniently by some diagram:
    - each element of V_G as a dot, the vertex,
    - and each element of E_G as a line segment, the edge, between two vertices.
  - So, V_G is called the set of vertices, and E_G, the set of edges.

Note the differences between Geometric diagrams and *Topological* diagrams.

A definition of a graph ***does not*** depend on the diagram.

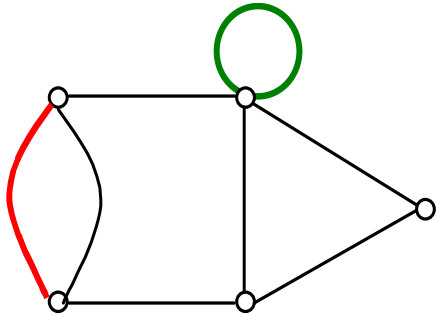
Relations in Graph

- Relation defined from edge set to the Cartesian product of vertex set
  - Incidence
- Relation defined on the set of vertex
  - Adjacency

Considering the relations above, a graph can be represented by matrix.

Simple Graph

- Multiple edges and ring
  - If φ is not a injection, that is, exist e_i, $e_j \in E_G$, $e_i \neq e_j$, but $\varphi(e_i)=\varphi(e_j)$, then e_i, e_j是are called multiple ***edges***.。
  - For any $e_i \in E_G$, if $\varphi(e_i)=\{v_i, v_i\}=\{v_i\}$, then e_i is called a ***ring***.
- Simple graph
  - A graph without ring and multiple edges is called a ***simple graph***.



(In a simple graph, an edge can be denoted as $uv \in E$.

Degree of Vertex

- Degree of vertex
  - $d_G(v)$ = number of edge incident to v

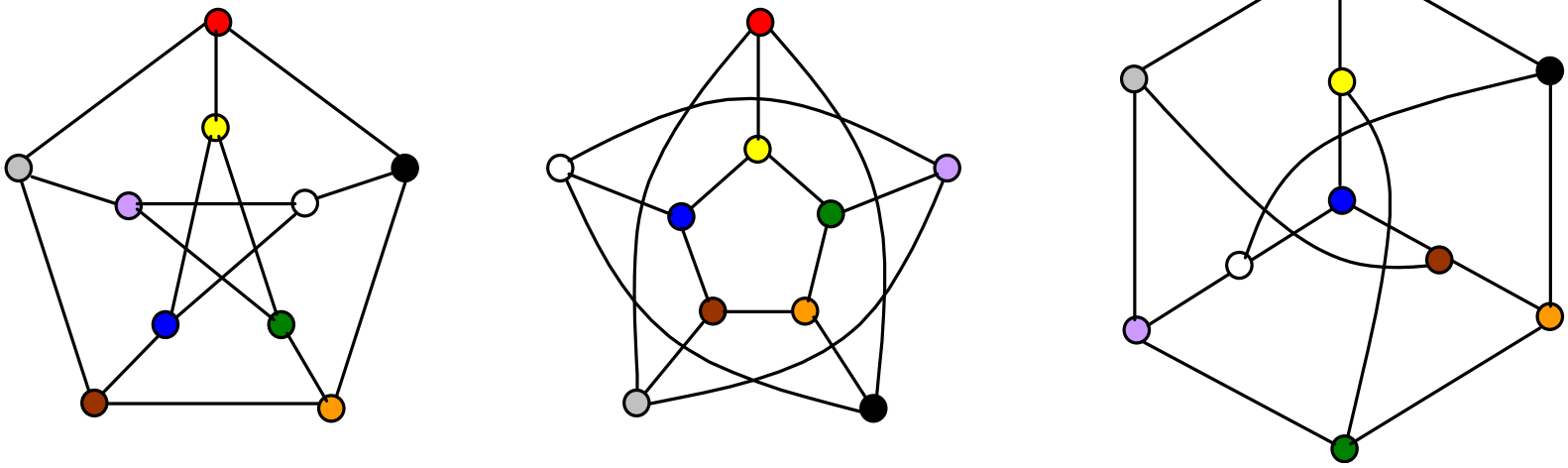
    $d_G(v)$ must be a nonnegative integer.
  - $\Delta(G)$ and $\delta(G)$
- Numerical characteristics of the degree of vertex
  - Sum of degree of all vertices in a graph is even.
  - $\sum_{i=1}^{n} d(v_i) = 2m$ m is the number of edges in the graph.
- The number of vertices with odd degree must be even.

Subgraph

- Let G=<V,E>, G'=<V',E'>, if V'⊆V, E'⊆E, then G' is called a subgraph of G.
- If V'⊂V, ***or*** E'⊂E, then G' is a proper subgraph.
- If V'=V, then G' is a spanning subgraph.

Isomorphic Graphs

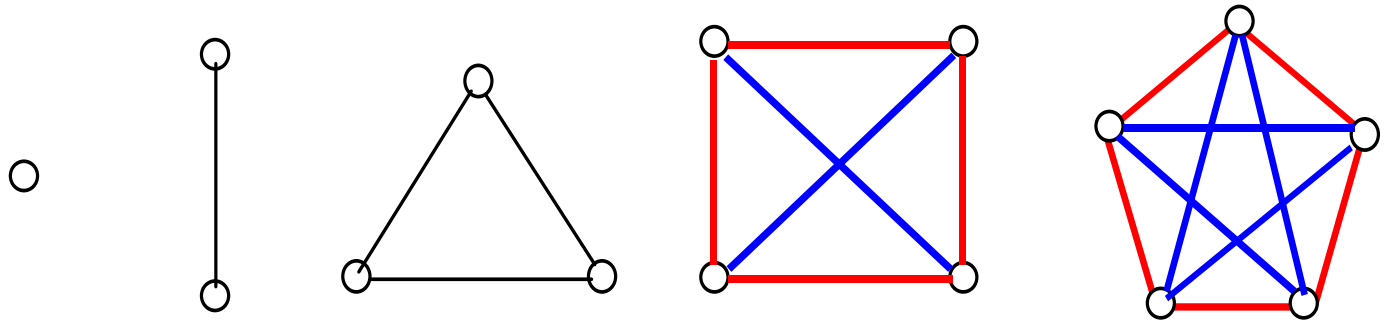
- The following three graphs are isomorphic (Petersen Graph)
  - The correspondences under the isomorphism are denoted by colors:



Complete Graph

- A graph is a complete graph if and only if any two of its vertices are adjacent.
  - In the sense of isomorphism, for a given n, there is exactly one complete graph, denoted as K_n, where n is the number of vertices in the graph.
  - In K_n, for any vertex v, $d(v)=n-1$.
  - The number of edges in K_n is exactly $n(n-1)/2$.
  - If G is a simple graph, then the number of its edges $m \le n(n-1)/2$

Complete Graph K_n(n=1,2,3,4,5)

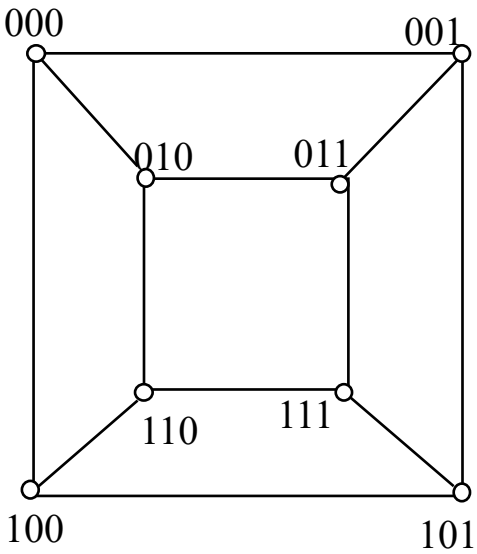


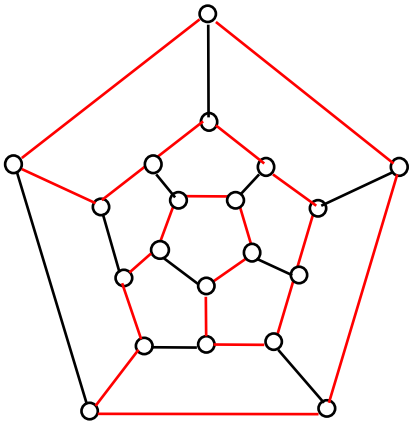
K_1 K_2 K_3 K_4 K_5

- A ***graph*** G and its ***complement graph*** have the same set of vertex, and their edges set constitute a partition of the edge set of the complete graph with the same vertex set.

Regular Graph

- Two typical regular graph:





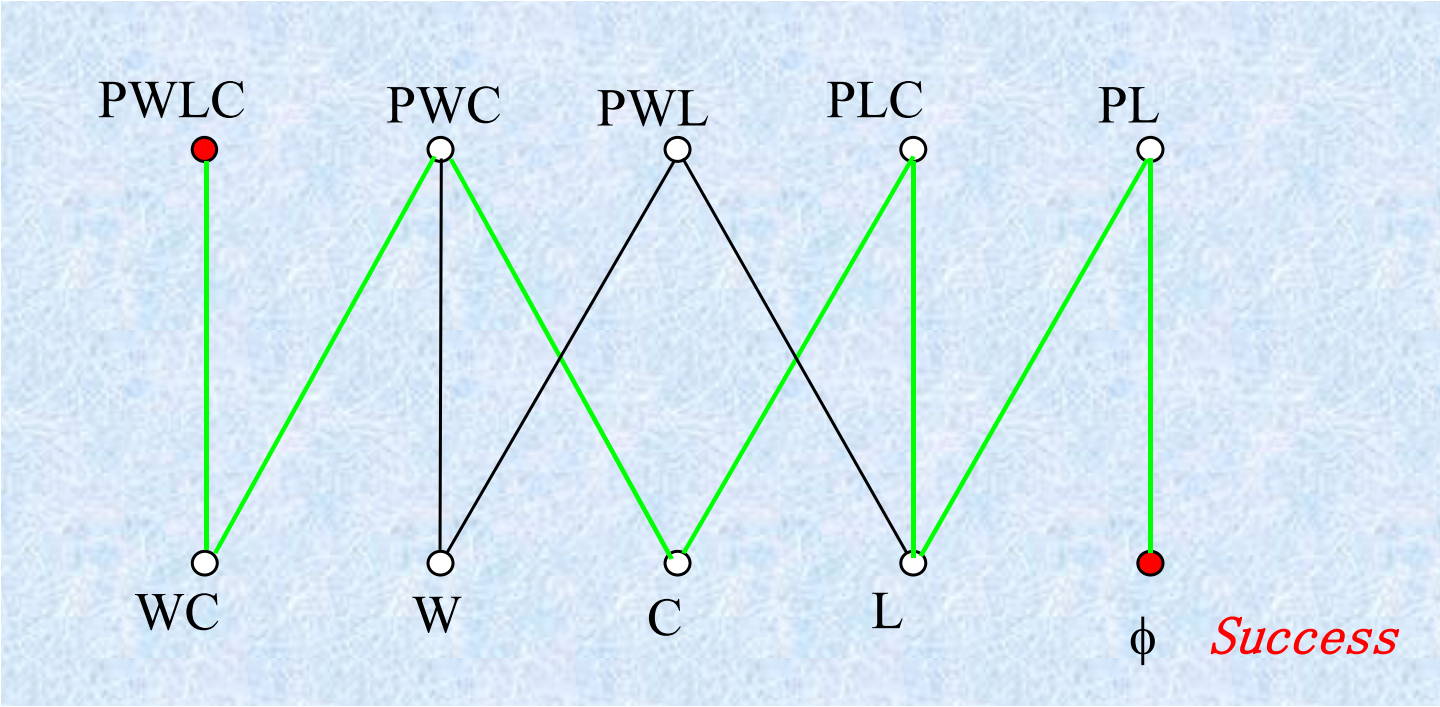
- Note: in the graph on the left, each vertex is a binary number with the same number of bits.

Problem of “Crossing River”

- Problem: A person(P), a wolf(W), a lamb(L) and a cabbage(C) will cross a river by a boat which can carry any two of them once. Wolf and lamb, or, lamb and cabbage, cannot stay together without the person present. Remember that only the person can run the boat.
- Graph model: vertex: “Status of Leaving Bank”， $uv \in E_G$ if and only if the transition from status u to v can be achieved by one allowed “cross” operation.
- Starting status: “PWLC”, ending status: empty.
- Solution: Finding a shortest path as possible from the starting status to the ending status.

Solution to the “Crossing River”

- Note: There are 16 combinations of (P,W,L,C), but only 10 status are possible.

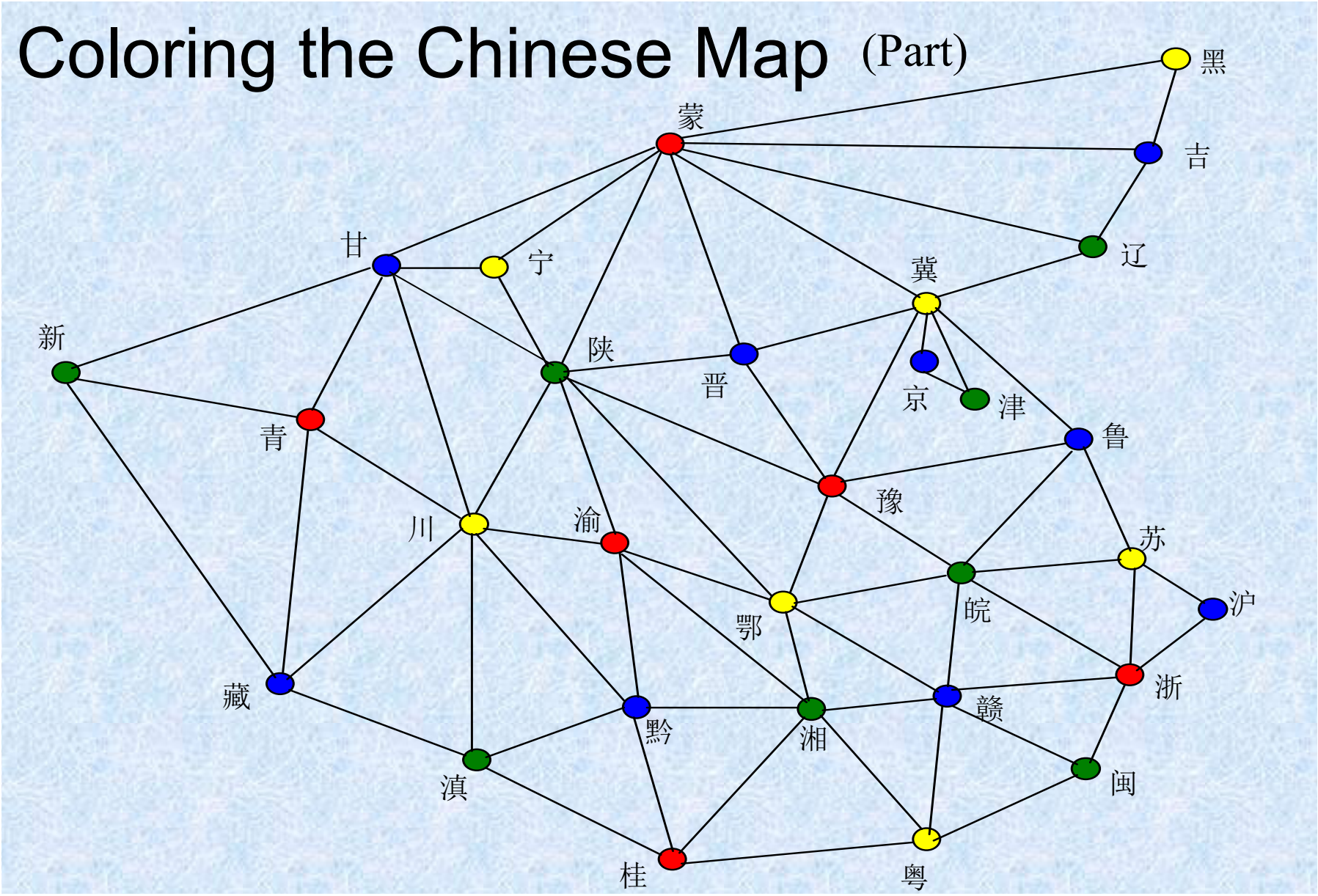


Scheduling the Exams

- Problem: Scheduling exams for different coursed. With no limitation on the space, we want to arrange all exams within a time span as short as possible. Any two course which is taken by one student cannot be scheduled in the same time slot.
- Graph model: vertex: course; edge: $uv \in E_G$ if and only if no student has taken both of courses u,v.
- Solution: Coloring the vertices with different colors such that no adjacent vertices are with the same color. If the coloring can be achieved using as few as k colors, then the smallest number of the time slots needed for the exams is k.

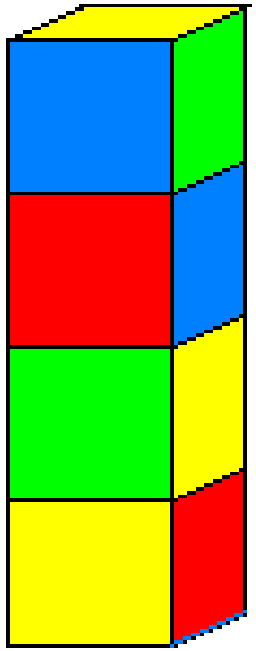
(The four-color problem is just a special case of this problem.)

Coloring the Chinese Map (Part)



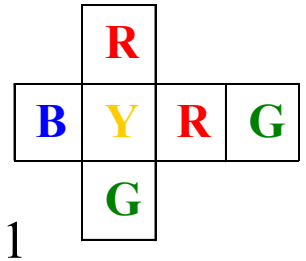
Magic Pole

- A “Magic Pole” consists of 4 cubic arranged as the right.
- Each of the 6 face of a cubic is colored by a color in {Red, Yellow, Blue, Green}
- Requirement: Each side of the “Magic Pole” show 4 different colors.

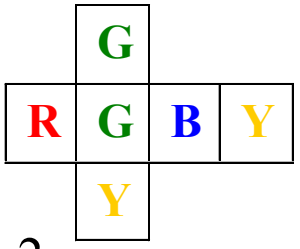


Problem Inputs

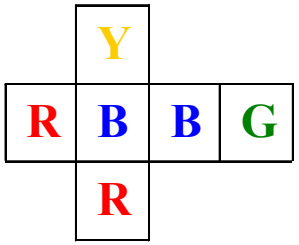
- A special pattern of color is showed on the right, which defines a specific problem input for “Magic Pole”.
- Problem: Does the solution exist? How to find it?



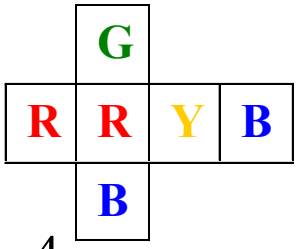
1



2

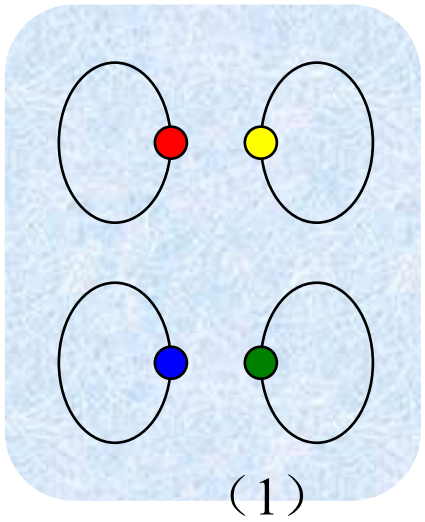


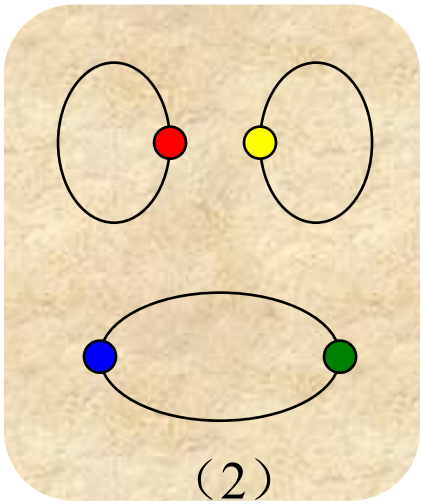
3

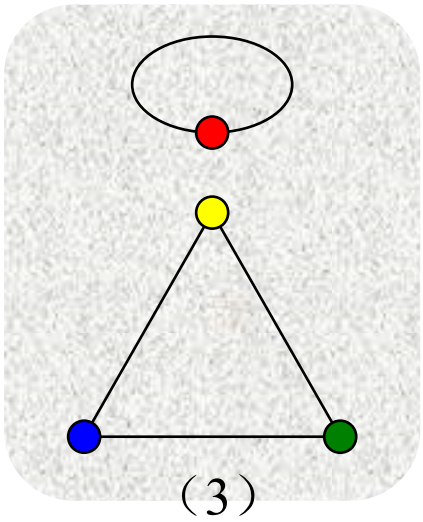


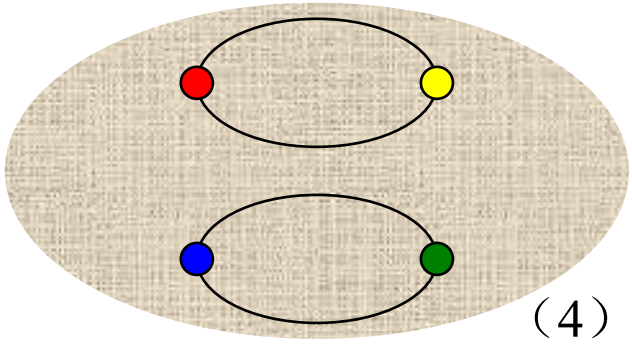
4

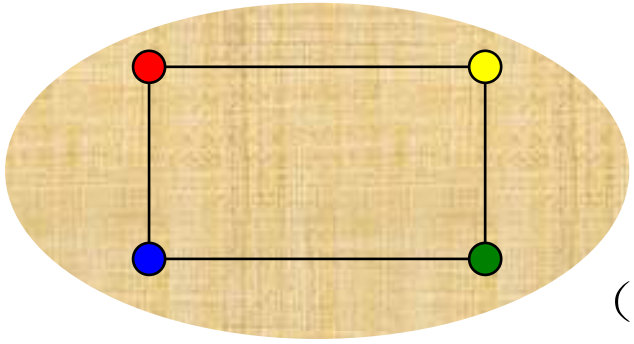
Possible Solution “Patterns”





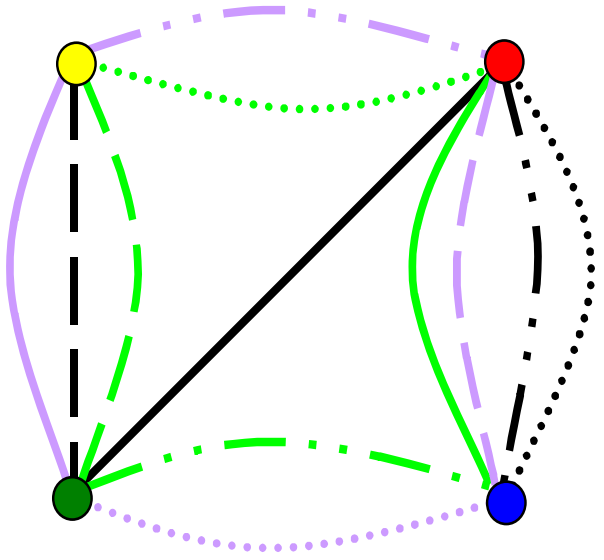


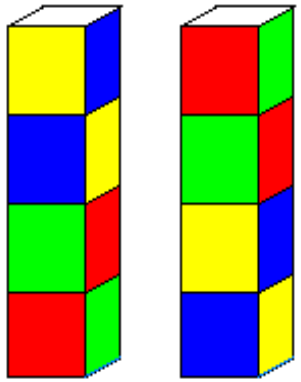




The Solution

- Graph for solving “Magic Pole”: For any two vertices u,v(color), $uv \in E_G$ if and only if u,v are opposite colors in a cubic.
- Front and back of the solution showed on the right.
  - Legend:
  - 1：
  - 2：
  - 3：
  - 4：





Path

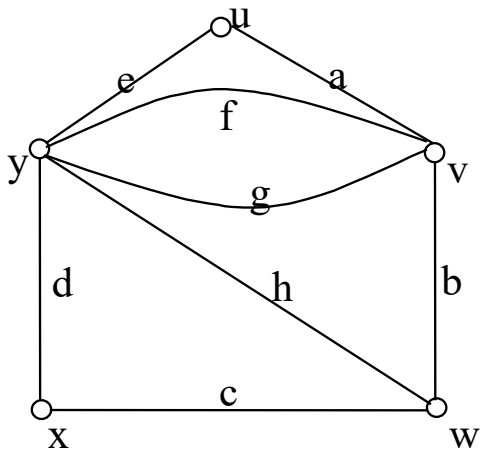
- Definition:
  - (v_0,v_k)-path in a graph G is a nonempty sequence: $v_0e_1v_1e_2v_2\ldots v_{k-1}e_kv_k$, among which $v_i \in V_G$, $e_i \in E_G$, and the two endpoints of e_i are v_{i-1} and v_i,(i=1,2,…,k)。
- Simple path:
  - $v_0e_1v_1e_2v_2\ldots v_{k-1}e_kv_k$, satisfying $\forall i,j,\ i\neq j \Rightarrow v_i \neq v_j$

Circuit (Cycle)

- Circuit is a path of which the starting point and the ending point are identical.
  - $v_0e_1v_1e_2v_2\ldots v_{k-1}e_kv_k$, satisfying $v_i=v_j$

Note: in a simple graph, the denotation of a path can be simplified as $v_0v_1v_2\ldots v_{k-1}v_k$,

- Examples:
  - Path: uavfyfvgyhwbv
  - Path: wcxdyhwbvgy
  - Simple path: xcwhyeuav
  - Circuit: ueyhwbvau



Existence of Simple Path

- If there is a (u,v)-path($u \neq v$) in graph G，then there must be a simple (u,v)-path
  - Proof:

    Let $W= v_0e_1v_1\ldots v_{k-1}e_kv_k$, is a (u,v)-path, where $v_0=u$, $v_k=v$。

    If $\forall i,j$, $i \neq j \Rightarrow v_i \neq v_j$, then W itself is a simple path.

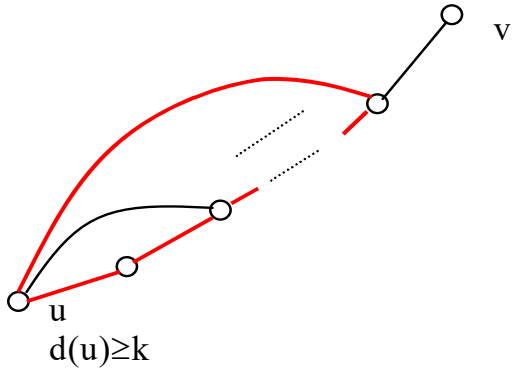
    Otherwise, exist $i,j(0 \leq i < j \leq k)$, such that: $v_i=v_j$

    That is $W=v_0e_1v_1e_2v_2\ldots v_{i-1}e_iv_i\ldots v_{j-1}e_jv_je_{j+1}\ldots v_{k-1}e_kv_k$,

    So, W‘= $v_0e_1v_1e_2v_2\ldots v_{i-1}e_iv_ie_{j+1}\ldots v_{k-1}e_kv_k$ is still a (u,v)-path，but the number of pair of identical vertices in W’ decreases

    Note: W is a finite sequence, repeat the above procedure, we can get a simple (u,v)-path.
  - If there are *n* vertices in G, the largest length of a simple path in G is *n*-1.

Least Vertex Degree and Circuit



- Let G be a simple graph if d_G=k (k>1), then there must be a simple circuit with the length at least *k*+1 in G.
  - Proof:
  - Let P is any path in G, its ending points are u,v. If both u,v are not adjacent to any points outside P, then P is called a maximal path.
  - *Note: with any graph with edges, we can always find a maximal path beginning from any edge.*
  - Since d_G>1, there must be a maximal path (v_0, v_1, v_2,…,v_{m-1}, v_m), then v_0 is adjacent to at least k different vertices on the path. So, among the k-1 vertices, the farthest from v_0 with a subscript not less that k. Let it be v_t, then (v_0, v_1,…, v_t, v_0) is a simple circuit with the length at least k+1.

Reachability

- Definition: $R_c \subseteq V_G \times V_G$, $\forall u,v \in V_G$, $\langle u,v \rangle \in R_c$ if and only if there is a (u,v)-path in G.
- We assume there is a path of length 0 from *v* to *v* for any vertex *v*.
- Obviously, R_c is an equivalence.
- In fact R_c is the transitive closure of the adjacence relation, R_a, on V_G.

Connected Graph and Components

- Reachability relation is a equivalence relation, the equivalence is a component of G.
- If there is only one component in G, then G is a connected Graph.
- ***Graph G is connected if and only if $\forall u,v \in V_G$, there is a uv-path in G.***

Connectivity of Complement Graph

- Let G be a simple graph, then G or G‘, the complement graph of G, is a connected graph.
  - Proof:

  Assuming G is disconnected, for any $u,v \in G'$:

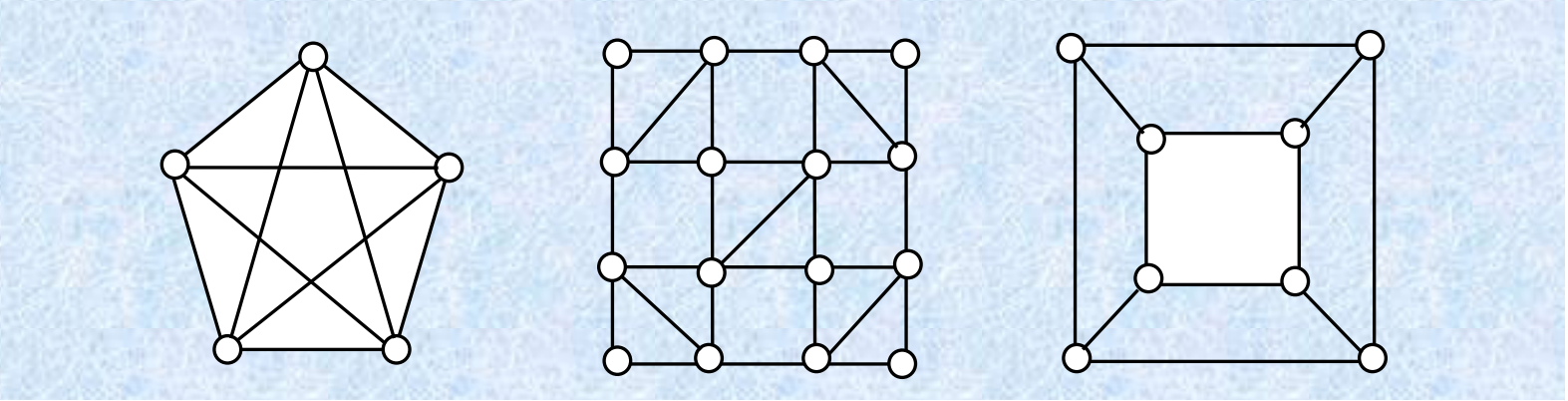
  if $uv \notin E_G$, u and v are adjacent in G‘;

  if $uv \in E_G$, since G is disconnected, there must be some vertex w, w is not in the component which u,v belong to. So $uw \notin E_G$, $vw \notin E_G$, so, $uw \in E_{G'}$, $vw \in E_{G'}$, that is, (u,w,v) is a uv-path in G‘.

  G‘ is connected.

Euler Path and Euler Circuit

- An Euler path in G is a path which passes each edge in G exactly once.
- An Euler circuit in G is a circuit which passes each edge in G exactly once.
- An Euler graph is a graph which contains a Euler circuit.
- A Semi-Euler graph is a graph which contains a Euler path, but not a Euler circuit.



Determination of Euler Graph

- Three equivalent statements for a nontrivial connected graph G:

  (1) G is an Euler graph.

  (2) For any vertex v in G, $d(v)$ is even.

  (3) All the edges in G constitute one or more simple circuits without common edges between them.

  - Proof:

  (1)⇒(2): Assuming that W is an Euler circuit in G, then $\forall v \in V_G$, $d(v)$ is 2 times of the number by which v appears on W.

Determination of Euler Graph(cont.)

(2)⇒(3): induction on the number of edges in G, m

- When m=1, G is a ring, the conclusion is true. Assuming that the conclusion is true for any $m \le k (k \ge 1)$
- If $m=k+1$, since $\delta_G > 1$, G contains simple circuit. Let C be a simple circuit, and $G'=G-E_C$, let G‘ contain s components, then in each component, every vertex has a even degree, and the number of edges is not larger than k. The inductive hypothesis applies, which leads to the conclusion.

(3)⇒(1). induction on the number of simple circuit in G

- Obvious for d=1. Assuming that the conclusion holds for $d \le k (k \ge 1)$.
- If $d=k+1$, with some ordering on all simple circuit, let G‘=G-$E(C_{k+1})$， For all component of G’, the inductive hypothesis applies, that is, each component is an Euler circuit. It is easy to constitue an Euler circuit for G using all the circuits in components.

Fleury Algorithm

Basic idea: when we really draw an Euler circuit, all passed edges cannot be used again. So, at any moment in drawing, with all passed edges deleted, the remain edges must be in one component. 。

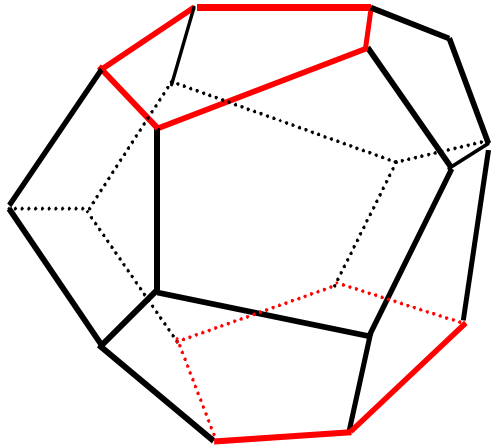
- Algorithm
  - Input: an Euler graph G
  - Output: a path $P = v_0e_1v_1e_2,\ldots,e_iv_ie_{i+1},\ldots,e_mv_m$, which includes each edge in E_G exactly once.
  - Procedure

1. Take $v_0 \in V_G$ randomly, let $P_0=v_0$;
2. Assuming $P_i=v_0e_1v_1e_2,\ldots,e_iv_i$, select e_{i+1} from $E_G-\{e_1,e_2,\ldots,e_i\}$ according to the following criteria:
   - (a) e_{i+1} is incident to v_i;
   - (b) If possible, e_{i+1} is not a cut edge in $G-\{e_1,e_2,\ldots,e_i\}$
3. Repeat step 2, until no firther step can be executed legally.

All Around the World

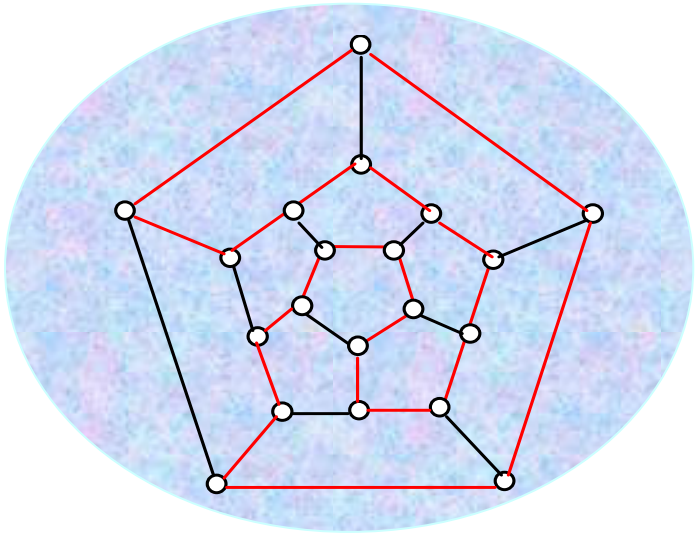
- In 1859, an Irish mathematician, Hamilton, introduced the game called “All around the world”

Each angular point of the dodecahedron represents a city all around the world. The player is required to start from a city, pass every city exactly once, along the edges, and return to the starting city finally.



Hamiltonian Path and Circuit

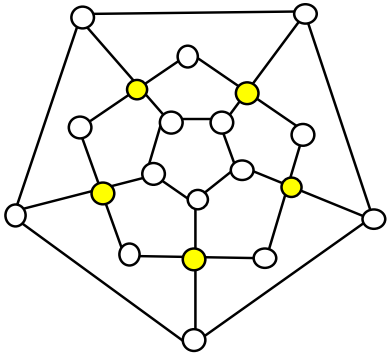
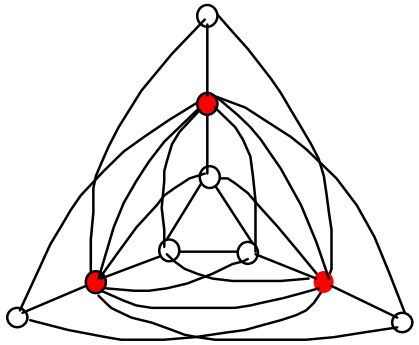
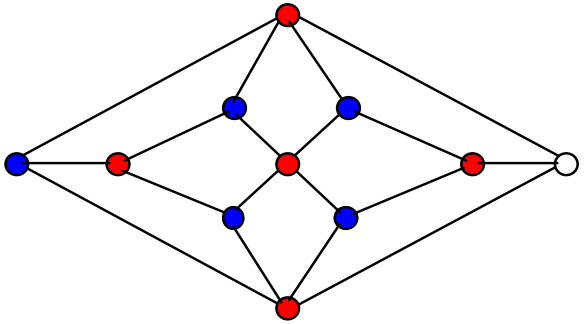
- In a graph G, a circuit is called a ***Hamiltonian circuit*** if and only if it contains all the vertices in G exactly once. If G contains a Hamiltonian circuit, G itself is called a ***Hamiltonian graph***.
- A Hamiltonian path is a simple path which contains all vertices exactly once.
  A semi-Hamiltonian graph contain a Hanmiltonian path, but not a Hamiltionian circuit.



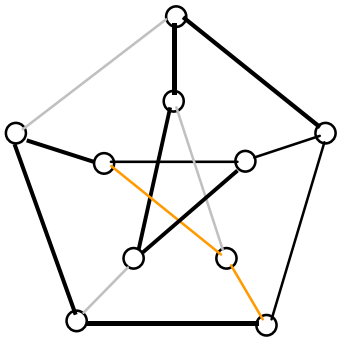
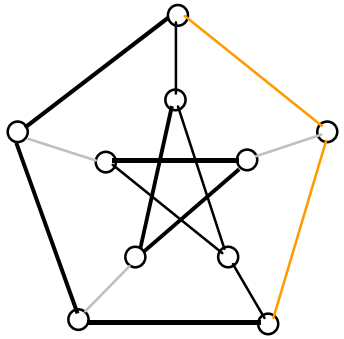
A Necessary Condition

- Observation:
  - Any Hamiltonian graph is in fact a simple circuit plus ***some*** linking some pairs of vertex on the circuit. 。
  - ***Deleting k vertices*** from a simple circuits results in ***at most*** k components.
  - ***Inserting edges*** between a pair of vertices not adjacent to each other will ***not*** increase the number of component.
- Conclusion:
  - Let G be a Hamiltonian graph, then for any nonempty subset V_1 of V_G, ***the number of components in $G-V_1$ is not larger than the number of vertices in V_1.***

Examples

- Necessary condition can only be used to determine that a graph is ***not*** a Hamiltonian graph.
  - In fact, Petersen (right) is not Hamiltonian graph.

Degree of Vertices and Connectivity

A graph must be connected if the lower bound of the sum of degrees of any vertex pair is large enough.

- G is a simple graph with no less than 2 vertices. G is connected if for any vertices *u*,*v* (not adjacent to each other):

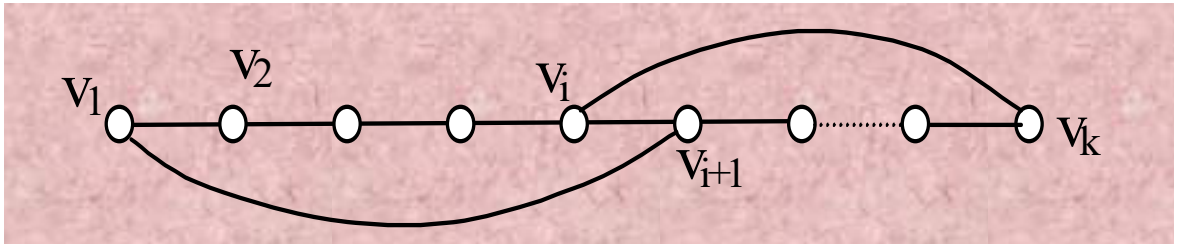
  $d(u)+d(v)\geq n-1$ (n is $|V_G|$) (condition *)

  - Proof:

  Assuming that G is disconnect, and G_1, G_2 are two of the components. For $x\in V_{G1}$, $y\in V_{G2}$, we have $d(x)+d(y)\leq(n_1-1)+(n_2-1)\leq n-2$ (n_i is $|G_i|$), contradiction.

Transform a Maximal Path to Cycle

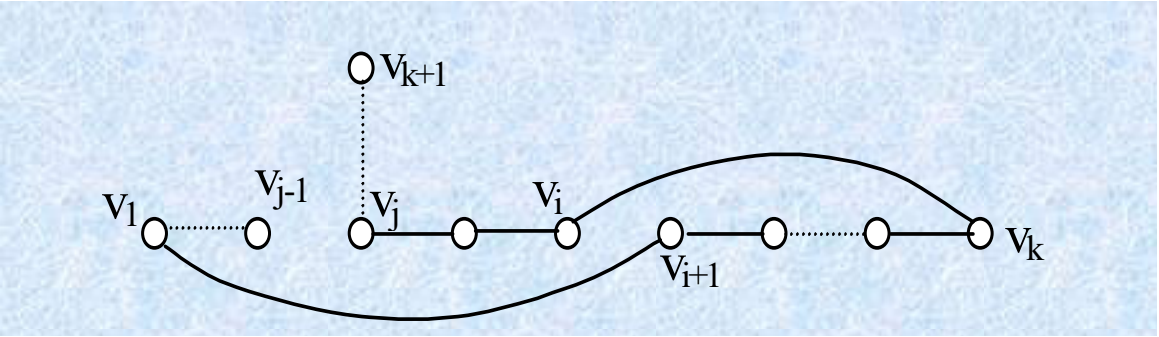
- If condition (*) holds for graph G, and let $\Gamma=v_1v_2\ldots v_{k-1}v_k$ be a maximal path not containing all vertices in G(k<n), then all vertices on Γ are on a simple circuit.
  - If $v_1v_k\in E_G$， conclusion holds.
  - Otherwise let $S=\{v_i|v_1v_{i+1}\in E_G\}$， $T=\{v_j|v_jv_k\in E_G\}$;
  - Note: $|S|+|T|= d(v_1)+d(v_k)\geq n-1$,
  - ： $v_k\notin S\cup T$, $\therefore |S\cup T|\leq k-1<n-1$, $\therefore |S\cap T|=|S|+|T|-|S\cup T|>0$, that is, $S\cap T$ is nonempty， let $v_i\in S\cap T$, then $v_{i+1}v_1\in E_G$， and $v_iv_k\in E_G$. So, $C=v_1\ldots v_iv_kv_{k-1}\ldots v_{i+1}v_1$ is a circuit containing all vertices on Γ.



Sufficient Condition for Semi-H Graph

Condition * *is a sufficient condition for semi-Hamiltonian graph.*

- If fact, if condition * holds for G, then the maximal path must be a Hamiltonian path.
  - Assuming that $\Gamma = v_1 v_2 \ldots v_{k-1} v_k$ is a maximal path, but k<n, then it can be transformed to a simple circuit C. Let v_{k+1} is one of the vertices outside of C. Since G is connected, there must be a path from v_{k+1} to the vertices in C. Assuming that the shortest of such paths meets Γ at v_j(not v_1,v_k, obviously). Then Γ ‘= $v_{k+1} \ldots v_j \ldots v_i v_k \; v_{k-1} \ldots v_{i+1} v_1 \ldots v_{j-1}$ is a extension of Γ, contradiction.



A Sufficient Condition for Hamiltonian Graph

- G is a simple graph with not less than 3 vertices. If for any vertices u,v not adjacent in G:

  $d(u)+d(v)\geq n$ (n is $|V_G|$)

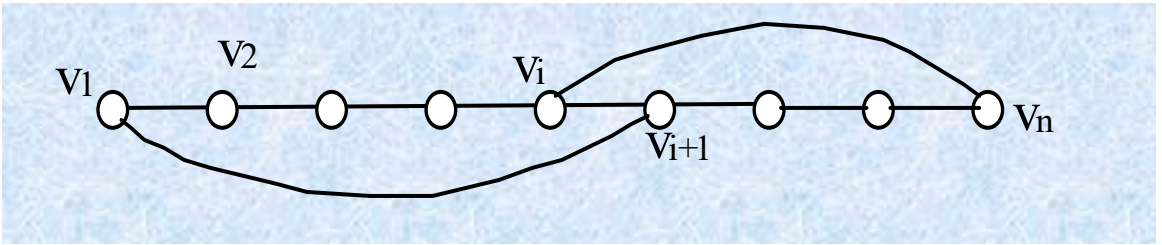
  Then G is a Hamiltonian graph.

- Proof:

  Obviously G is a semi-Hamiltonian graph. Let $\Gamma = v_1v_2\ldots v_{n-1}v_n$ be a Hamiltonian path in G.

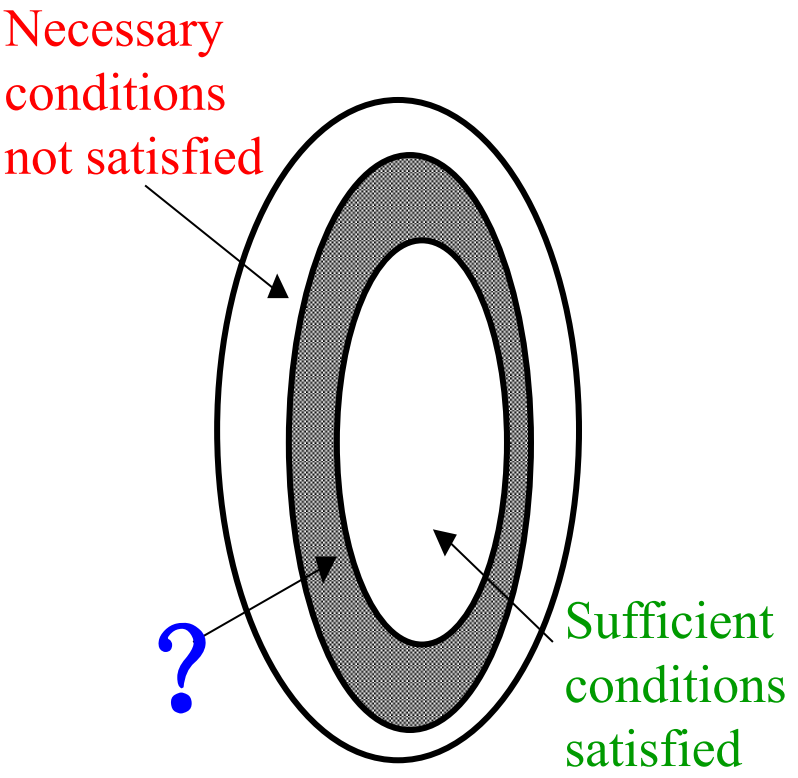
  If $v_1v_k \in E_G$, conclusion holds. ***Otherwise***, let $S=\{v_i|v_1v_{i+1} \in E_G\}$, $T=\{v_j|v_jv_k \in E_G\}$. Note: $|S|+|T|= d(v_1)+d(v_n)\geq n$,

  ∵ $v_n \notin S\cup T$, $\therefore |S\cup T|\leq k-1<n$, $\therefore |S\cap T|=|S|+|T|-|S\cup T|>0$, that is, $S\cap T$ is nonempty, let $v_i \in S\cap T$, then $v_{i+1}v_1 \in E_G$, $v_iv_n \in E_G$. So, $C=v_1\ldots v_iv_nv_{n-1}\ldots v_{i+1}v_1$ is a Hamiltonian circuit.



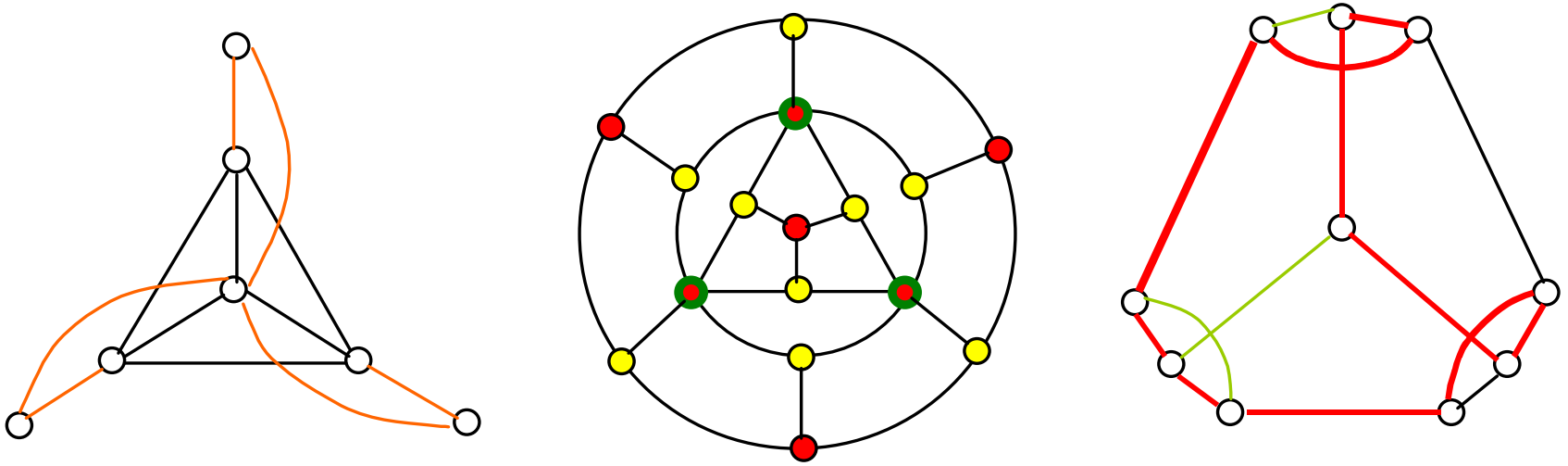
Gaps between Two Kinds of Conditions

- Determined case by case using "common senses"
  - For each vertex, exactly two of the incident edges are contained in the Hamiltonian circuit.
  - No proper sub-circuit can be contained in a Hamiltonian circuit.
  - Making use of symmetry
  - .....



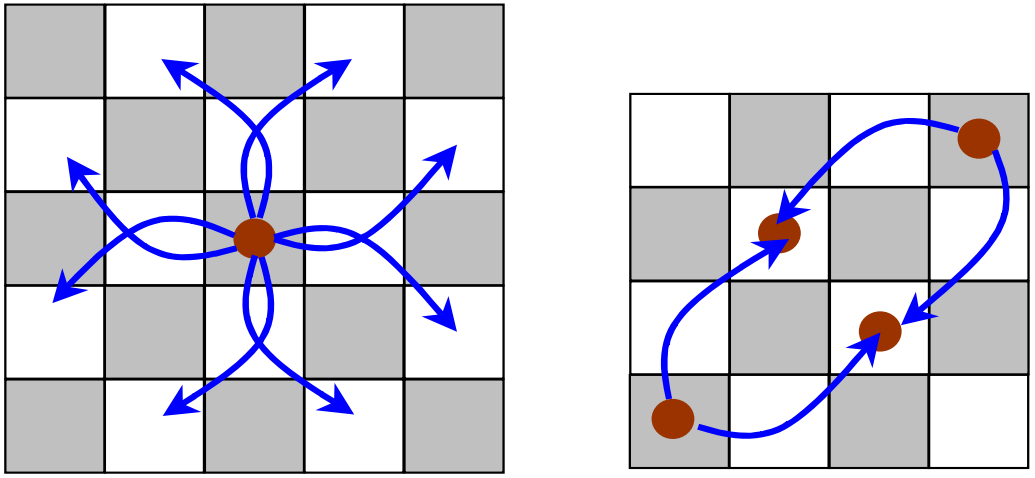
Examples

- Only the graph on the right is Hamiltonian



Hamiltonian Circuit on a Chess Board

- On a minimized 4×4 or 5×5 chess board, Knight cannot traverse each square exactly once and return to the starting square.



Traveling Sales Person (TSP)

- Problem: There are n cities with road of different distances between any two of them. A traveling sales person plan to traverse each city exactly once and return to the starting place. How should he choose the shortest traversal route?
- Mathematical model:
  - Construct a weighted graph G， with elements in V_G中 corresponding to cities, and those in E_G, to road between cities. The weights are the distances.
  - The solution is a Hamiltonian circuit with the least weight.
  - G, as a complete weighted graph, has n!/2 different Hamiltonian circuit. The problem is "which is the shortest?"

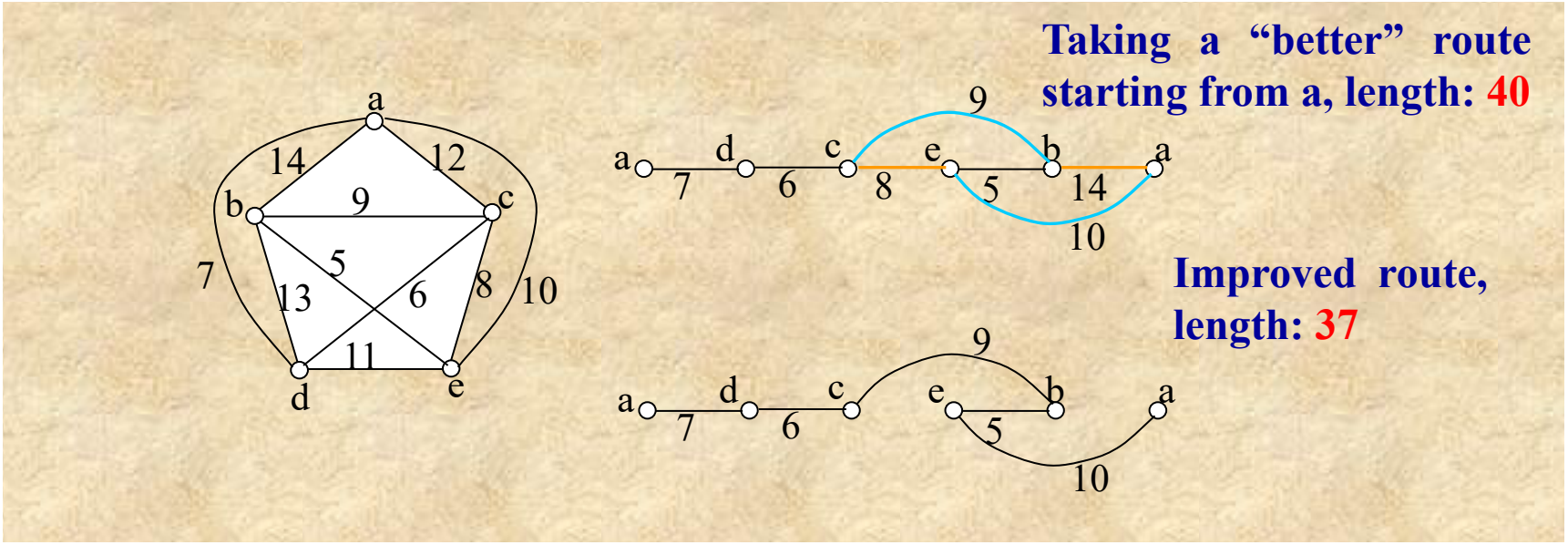
(For your reference: there are about $(1.21645\times10^{17})/2$ different candidate routes in a complete Hamiltonian graph with 20 vertices. If check mechanically, with the speed of 100,000/sec. it takes 20,000 years!)

TSP: an Approximate Algorithm

- The procedure

  (1) Find a route using "Greedy approach" – may not best.

  (2) Improvement: If, in current route, exists $W(v_i,v_j)+ W(v_{i+1},v_{j+1})< W(v_i,v_{i+1})+ W(v_j,v_{j+1})$, then replace v_iv_j and $v_{i+1}v_{j+1}$ with v_iv_{i+1} and v_jv_{j+1}



Home Assignments

- To be checked
  - Ex.8.1: 16-18, 27-29, 31-32
  - Ex.8.2: 4, 6, 9, 12, 13-14, 18, 19-25
  - Ex.8.3: 3, 6, 10, 14, 19-25