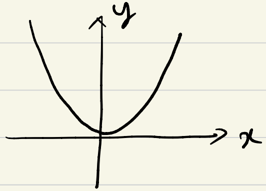
Ch 8. Graph Theory

§8.1. Introduction

In calculus, the graph of $y=x^2$ means

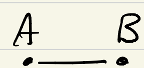


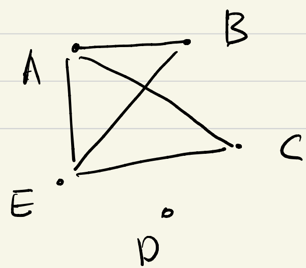
그래프 이론

In graph theory, a graph is an abstract object consisting of vertices 정점 and edges 엣지. Many situations can be described using graphs.

ex) Friendship among people.

A, B are friends $\leftrightarrow$





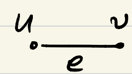
Def) A graph G consists of a set V of vertices and a set E of edges such that $e \in E$ is an $\boxed{\text{unordered pair}}$ of vertices.

ex) $G=(V,E)$, $V=\{A,B,C,D,E\}$.

$E=\{(A,E),(A,B),(A,C),$
$(B,E),(E,C)\}$.

"unordered" means $(A,B)=(B,A)$

If $e=(u,v)$, then we say that



인접한

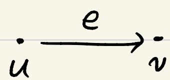
u and v are adjacent,

u, v are incident with e.

연결된

Def) A <u>directed graph</u> (digraph) G consists of a vertex set V and an edge set E such that each $e \in E$ is an ordered pair of vertices.

$e = (u, v)$

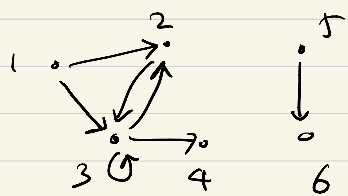


"ordered" means $(u,v) \neq (v,u)$

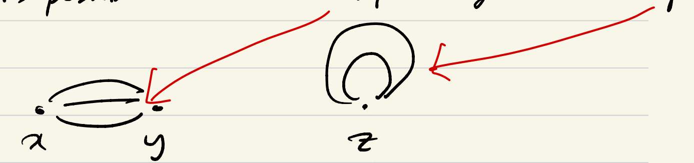
ex) $G = (V, E)$ directed graph

$V = \{1, 2, 3, 4, 5, 6\}$

$E = \{(1,2), (1,3), (2,3), (3,4), (3,2), (3,3), (5,6)\}$

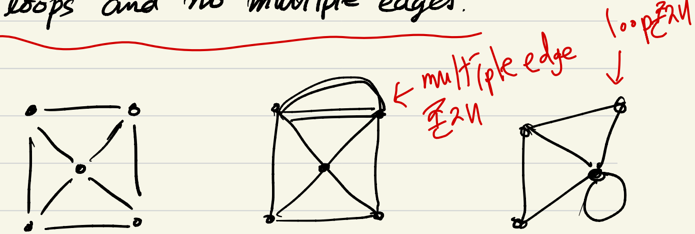


It is possible to have multiple edges and loops.



Def) A <u>simple</u> graph is a graph with no loops and no multiple edges.

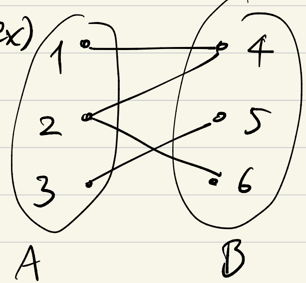
ex)

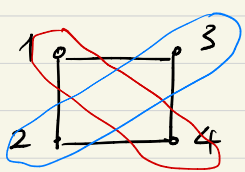


simple. not simple not simple

Def) A graph $G=(V,E)$ is called bipartite (이분적인) if there are subsets $A,B \subseteq V$ such that $A \cap B = \phi$ and $A \cup B = V$ and every $e \in E$ is an edge (u,v) where $u \in A$, $v \in B$.

하나는A 하나는B

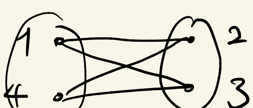
ex)  is bipartite

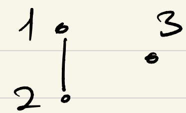
ex) 

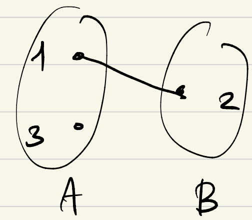
A,B를 어떻게 고르는지에 따라 bipartite가 된다

$G=(V,E)$
$V=\{1,2,3,4\}$
$E=\{(1,2),(1,3),(2,4),(3,4)\}$

$A=\{1,4\}$, $B=\{2,3\}$ $\Rightarrow$ bipartite

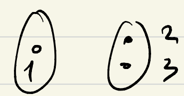


ex)  is bipartite.



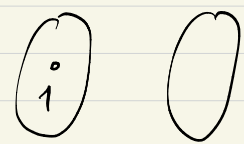
edge가 없다면 항상 bipartite

ex) 1 2 3 is bipartite.

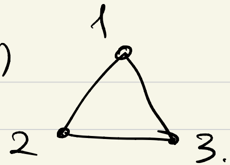


하나여도 bipartite

ex) 1 is bipartite.



$A=\{1\}$ $B=\phi$

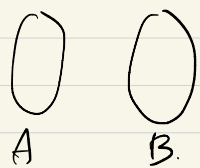
ex) 

is not a bipartite graph.

$G=(V,E)$.

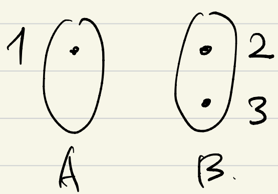
$V=\{1,2,3\}$

$E=\{(1,2),(2,3),(3,1)\}$.



Suppose $A\cap B=\phi$, $A\cup B=V$

We may assume $1\in A$. ← 일반성을 잃지 않음



Since $(1,2)\in E$, we must have $2\in B$.

" $(1,3)\in E$, " $3\in B$

But $(2,3)\in E$, $2,3\in B$, so

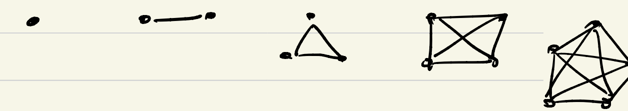
we cannot find A, B satisfying the conditions in the def of bipartite graph. 조건에 (2,3)이 있기에

Therefore G is not partite.

Def) $\boxed{K_n}$ is the complete graph on n vertices. (완전 그래프)

(every two distincts vertices are connected by an edge.)

e.g. K_1 K_2 K_3 K_4 K_5



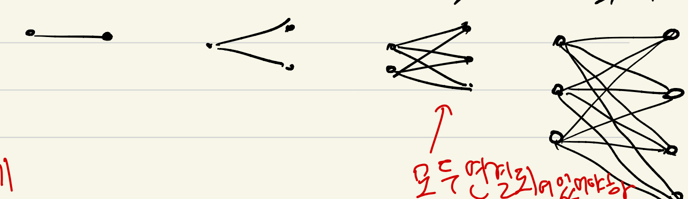
Def). $\boxed{K_{n,m}}$ is the complete bipartite graph (완전 이분 그래프)

on $V=V_1\cup V_2$ (disjoint union)

$|V_1|=n$, $|V_2|=m$.

where $E=\{(u,v) : u\in V_1, v\in V_2\}$.

e.g. $K_{1,1}$ $K_{1,2}$ $K_{2,3}$ $K_{3,4}$



↑ 모두 연결되어 있어야 함

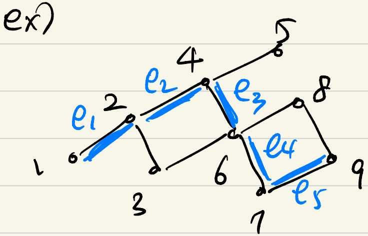
§8.2. Paths and Cycles

Def) $G=(V,E)$ graph.

A path from u to w of length n is a sequence $(v_0, e_1, v_1, e_1, v_2, \ldots, v_{n-1}, e_n, v_n)$ such that $v_0, \ldots, v_n$ are vertices, $e_1, \ldots, e_n$ are edges, $v_0 = u$, $v_n = w$, and $e_i = (v_{i-1}, v_i)$ $\forall i = 1, \ldots, n$.

Sometimes if the meaning is clear, we will omit e_i and write $(v_0, v_1, \ldots, v_n)$.

ex)



$(1, e_1, 2, e_2, 4, e_3, 6, e_4, 7, e_5, 9)$

edge개수

is a path from 1 to 9 of length 5.

$(1, 2, 4, 6, 7, 9)$.

$V \to V$

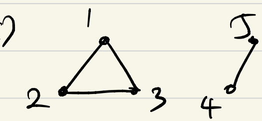
Note: (v) is a path from v to v of length 0.

연결된

Def) $G=(V,E)$ is connected if for any two vertices u, v there is a path from u to v.

ex) The graph in the prev ex is connected.

ex)



is not connected. (disconnected).

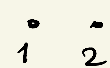
There is no path from 1 to 4 1→4 가는 길이 없기에

ex)

is connected? Yes.

There is a path from 1 to 1 (1).

ex)

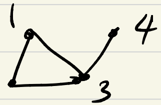


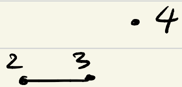
is disconnected.

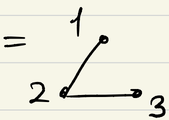
Def) $G=(V,E)$

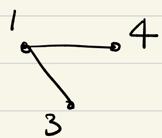
A subgraph of G is a graph $H=(V',E')$ such that $V' \subseteq V$, $E' \subseteq E$.

ex)

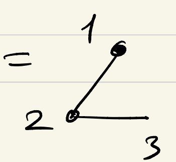
$G =$ 

$H_1 =$  → Subgraph

$H_2 =$  → subgraph

$H_3 =$  → not a subgraph

$(1,4) \notin E$.

$H_4 =$ 

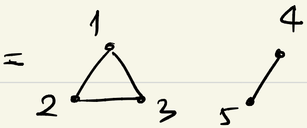
$V_4 = \{1,2\}$

E는 V의 pair인데 V에 3이 없기에

$E_4 = \{(1,2),(2,3)\}$.

→ not a subgraph.

not even a graph.

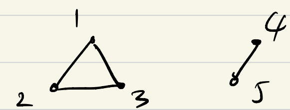
ex) $G =$ 

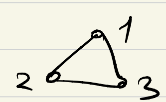
G is not connected.

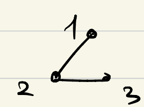
Def) $G=(V,E)$.

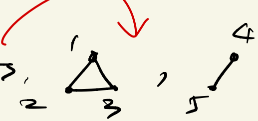
A (connected) component of G is a subgraph H of G consisting of all vertices and all edges of G that are contained in any path beginning at a fixed vertex v.

하위그래프 = 고정정점 v에서 갈 수 있는 모든 E와 V로 구성된 것

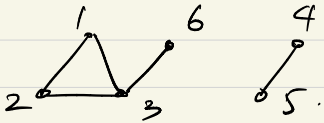
ex) 

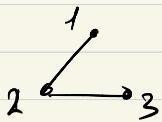
$v=1$. Then the component containing v is 

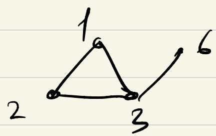
But  is not a component of G.

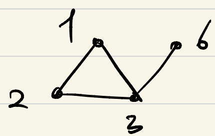
So, G has two components. 

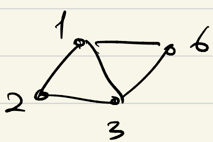
Note: A component can be understood as ①조건 the largest ②조건 subgraph that is connected 연결된 and containing a given vertex. 정점모두포함

G = 

 is not a component of G

because  is larger subgraph.

H =  is a component of G.

Warning:  is connected and larger than H.

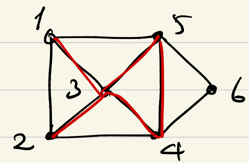
But this is not a subgraph of G.

Def) $G=(V,E)$ graph, $u,v \in V$.

A simple 단순경로 path (단방향) from u to v is a path from u to v with no repeated vertices.

A cycle 사이클 is a path of nonzero length from u to u with no repeated edges.

A simple cycle is a cycle with no repeated vertices. (except for the starting and ending points)

ex) 

(1, 3, 5, 4, 3, 2) is a path which is not simple.

repeat

(1, 3, 4, 2) is a simple path.

(1, 3, 5, 4, 3, 2, 1) is a cycle. (not simple).

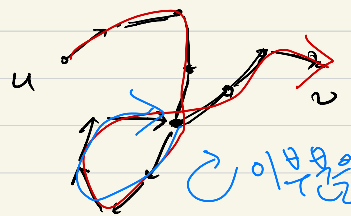
(1, 3, 4, 2, 1) is a simple cycle.

Prop $G=(V,E)$.

① If there is a path from u to v then there is a simple path from u to v.

② If there is a cycle from u to u then there is a simple cycle from u to u.

Pf) Idea:



이부분을 삭제하면 u→v의 simple path가 나옴

Or we can consider an extreme case.

증명 과정

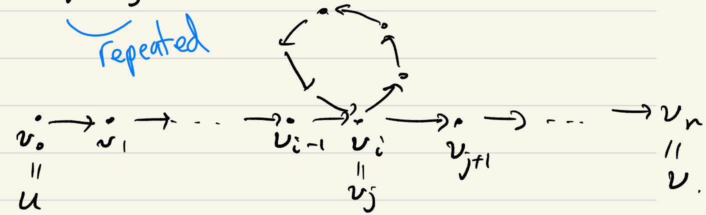
① Take a shortest path p from u to v.

We claim that p is simple.

Suppose p has repeated vertices.

$p = (u=v_0, v_1, v_2, \ldots, v_i, \ldots, v_j, \ldots, v_n = v)$.

$v_i = v_j$ repeated



Then $p' = (v_0, \ldots, v_i, v_{j+1}, \ldots, v_n)$ is a path from u to v which is shorter than p,

a contradiction.

So p has no repeated vertices

$\Rightarrow$ p is simple.

② Similar. □

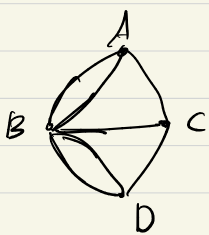
Def) $G=(V,E)$. $v\in V$.

The *degree* of v is the number of edges incident to v. ⇒ 정점 v에 연결된 E개수

The degree of v is written $\deg(v)$

(or in textbook $\delta(v)$.) 델타

ex).



G

$\deg(A)=3$

$\deg(B)=5$

$\deg(C)=3$

$\deg(D)=3$.

e.g.

$$\sum_{v\in V}\deg v = 3+5+3+3 = 14.$$

$$|E| = 7$$

Thm Let $G=(V,E)$.

Then

$$\sum_{v\in V}\deg v = 2|E|.$$ 모든 degree 합 = 2|E|

Pf) Every edge $e\in E$, $e=(u,v)$, 1개의 E는 2개의 v를 연결하기에

Contribute 1 to $\deg(u)$ and 1 to $\deg(v)$.

Therefore every edge contributes 2 in LHS.

$\Rightarrow$ LHS $= 2\cdot|E|$. $\square$

Cor The sum of degrees of all vertices is even

Cor The number of vertices of odd degrees is even.

Pf)

$$\sum_{v\in V}\deg v = \text{even}$$

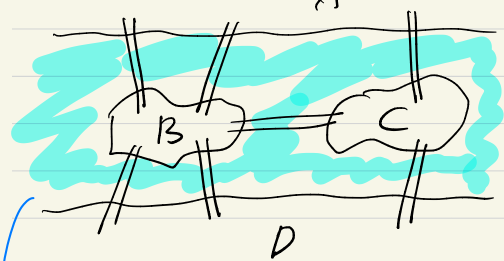
$$\| $$

$$\left(\sum_{\substack{v\in V\\ \deg v:\text{even}}}\deg v + \underbrace{\sum_{\substack{v\in V\\ \deg v:\text{odd}}}\deg v}_{\#\text{ terms is even.}}\right) = \text{even}$$

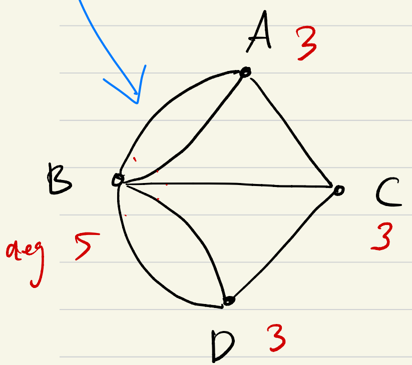
$\deg v$ 가 홀수인 애들이 짝수번 존재해야함 $\square$ ⇒ 그래야 짝수

쾨니히스베르크 다리 문제

ex) Königsberg Bridge Problem



Is it possible to walk over each bridge exactly once and return to the starting location?

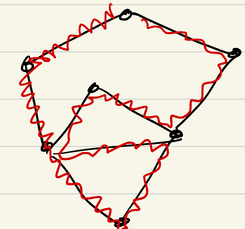


Is there a cycle that uses every edge exactly once?

오일러 사이클

Def) An Euler cycle of $G=(V,E)$ is a cycle that uses every edge exactly once and that visits every vertex.

ex)



논리적 동치

Thm. If $G=(V,E)$ has an Euler cycle then G is connected and every vertex has even degree.

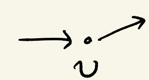
Pf). let $(v_0, e_1, v_1, e_2, \ldots, e_n, v_n)$ be an Euler cycle.
For any $u, v \in V$ we have $u = v_i$, $v = v_j$ for some i, j.
We may assume $i \le j$. Then there is a path from v_i to v_j, which is $(v_i, v_{i+1}, \ldots, v_j)$.

$\Rightarrow$ G is connected

Let $v \in V$. Then

$\deg v = 2$ (# times v appears in this Euler path).

($\because$ Once we visit v, we use two edges of v.)



정확히 1번만 지나가게 엣지가 2개(입/출)

Thm If G is connected and every vertex has even degree, then G has an Euler cycle.

=모든간선을 사용함

Pf) Fix a vertex $v \in V$. Take a $\boxed{\text{longest}}$ $\boxed{\text{path}}$ $p = (v_0, v_1, \dots, v_n)$ starting from v which do not contain repeated edges.

Claim : p is an Euler cycle. ⇒ ① 모든 엣지변색 ② 사이클

① We show that $\boxed{p \text{ is a cycle.}}$ $(v_n = v_0)$.

Suppose $v_n \neq v_0$. Then we used odd # edges incident to v_n. But $\deg v_n$ is even, so there are some edges incident to v_n that are not yet used. Then we can extend p by using one of these unused edges. This is a $\boxed{\text{contradiction}}$ to the assumption that p is longest.

② We show that p uses every edge in G.

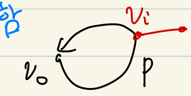
Suppose some edges are not used.

Let $G' = G - \{\text{edges in } p\}$.

Then every vertex in G' has even degree.

G; 짝수 차수 - p 해도 짝수임

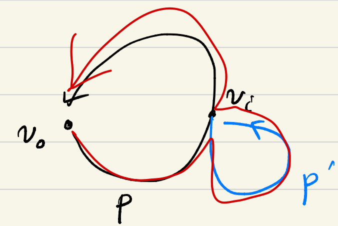
We can find an edge in E' incident to some v_i because otherwise G is not connected.



Suppose v_i has nonzero degree in G'.

Then by taking a longest path starting from v_i in G' we can find a cycle from v_i to v_i in G'.

$p' = (v_i, u_1, \dots, u_k, v_i)$.



p, p' 합친거

Let $p'' = (v_0, \dots, v_i, u_1, \dots u_k, v_i, \dots, v_n)$.

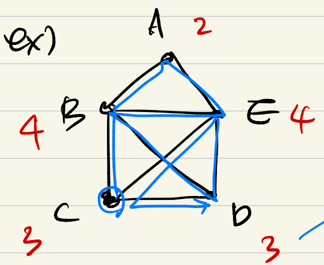
Then p'' is a cycle from v_0 to v_0 which is longer than p. ⇒ $\boxed{\text{contradiction.}}$ ⇒ p는 가장긴 경로임 (모든 간선 사용)

Therefore, every vertex v_i, we used all edges incident to v_i. Since G is connected there is no vertex outside p. ⇒ p is Euler cycle. ◻.

오일러 경로

Def) An Euler path of $G=(V,E)$ is a path from u to v $(u \neq v)$ that contains all vertices and uses every edge exactly once. ← 사이클이 아님

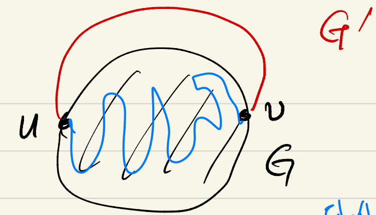
ex)



→ 오일러 사이클이 아님

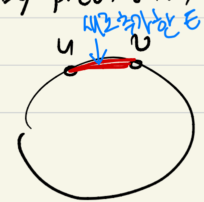
Thm G has an Euler path iff $\Leftrightarrow$ G is connected and # vertices of odd degree is 2.

Pf) $(\Rightarrow)$ let p be an Euler path from u to v. Then clearly G is connected.
Let $G' = G + (u,v)$. ← 추가



Then $p' = \underbrace{p}_{(u,\dots,v)} + (u,v)$ is an Euler cycle of G'. ← 다시 돌아옴
By the previous theorem, every vertex in G' has even degree.
Since $\deg_G(u) = \deg_{G'}(u) - 1$) u
$\deg_G(v) = \deg_{G'}(v) - 1$,) v
and $\deg_G(w) = \deg_{G'}(w)$ $\forall w \neq u, v$,
G has two vertices of odd degree, namely u, v.

$(\Leftarrow)$ let u, v be the two vertices of odd degree.
Let $G' = G + (u,v)$. Then G' is connected and every vertex in G' has even degree.
By prev. thm, G' has an Euler cycle.



By removing (u,v), we get an Euler path from u to v in G. $\square$

§8.3. Hamiltonian cycles and traveling salesperson problem.

해밀턴 경로

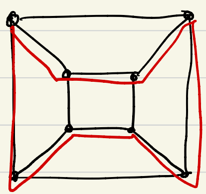
Def) A Hamiltonian cycle of G is a cycle that uses every vertex exactly once. (except the starting and ending vertex).

Euler cycle : uses every edge once

Hamiltonian cycle : " every vertex " .

ex)

G =



red cycle is a Hamiltonian cycle.

Note Unlike Euler cycles, finding a Hamiltonian cycle is not an easy problem → 쉽지않다!

This is an NP-problem.

Roughly speaking

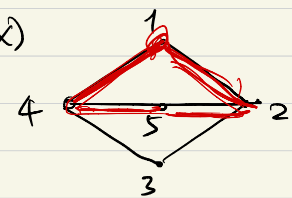
P-problem : a problem that can be solved in poly time. 상대적으로 빠르게

NP-problem : a problem whose solution can be checked in poly time.

답이 맞는지 확인

Famous open problem : P = NP.

ex)



This graph has no H.C.

pf) Suppose ∃ H.C.

Since 5 must be visited and it is connected to two edges, these two edges must be used.

Similarly, 14, 12 must be used.

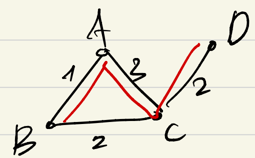
Then we already get a cycle but without 3. 또, 3을 지나려면 다른 점 점을 2번 지날수밖에..

So No H.C. ∎

가중치 그래프

Def) A weighted graph is a graph $G=(V,E)$ in which every edge has a number (weight).

e.g.



path $B \to A \to C \to D$ has wt $= 1+3+2=6$.

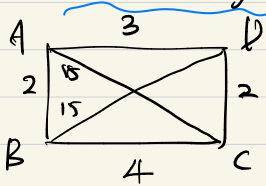
경로의 가중치 = 가중치 합

Def) The weight of a path or a cycle in a weighted graph is the sum of the weights of the edges in it.

ex) TSP (Traveling Salesperson Problem).

$G=(V,E)$ weighted graph.

Find a Hamiltonian cycle of G with minimum weight. 이게 정답



$A \to C \to D \to B \to A$

15 2 15 2

wt = 34.

$A \to B \to C \to D \to A$

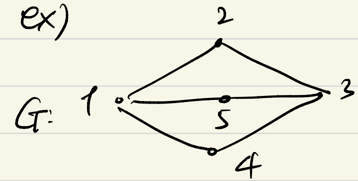
2 4 2 3

wt = 11.

HC.

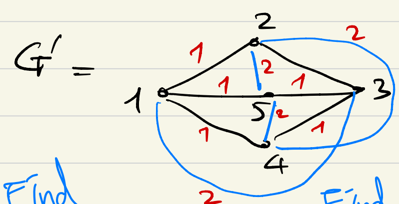
Note: TSP is related to finding a HC.

ex)



Does G have a HC?

$\Downarrow$ Tsp 문제로 변환



Add new edges to G so that we get K_n. 완전그래프

original edges wt = 1

new " = 2.

Find G has HC $\Longleftrightarrow$ Find minimal HC of G' has weight n (= # vertices). 정점수

못찾는다면 G는 H.C를 갖지 않는다.