Mathematics of Computer Science. - M.I.T. opencourseware

Abstract:
In this lecture we learned the Euler tour(trail) that is called '한붓그리기' in Korea and directed graph. directed graph is the graph that edge have head and tail like vector which play a role of bridge to linear algebra (Set). Lastly we got Tournament graph subset of DG that walk every vertex.

For the English:
peck, implication

Lec 10. GRAPH THEORY III:
- Euler tour
DEF: a walk that transfer every edge onec and starts and finishes at the same vertex
THM: connected graph has an Euler tour iff every vertex has even degree
Implication: assume G= (V, E) has an euler tour, since every edge in E is traversed once
deg(u) = # times u appears in tour v0 ~ vk times 2 (enter & exit)

Implication: for G(V, E) assume deg(v) is even for all v∈V, let W(walk) : v0 ~ vk be the longest walk that traverses no edge more than once
2) vk = v0, otherwise
1) vk-u not in W: longer one ->contradiction
All edges incident to vk are used in W

3) vk = v0, otherwise vk has odd degree W
by 1): vk has odd degree in G => contradiction
Suppose W is not an Euler tour, G is connected: so ∃edge is not used in W but it is incidnet to some vertex in W
-> let u-vi be this edge -> longer walk -> contradiction

- Directed Graph
the graph's edge has head and tail that shows direction like vector
THM: let G = (V, E) be an n-node graph with V = {v1 ~ vk} let A ={aij} denote the adjacent matrix for G, that is aij { 1 if vi -> vj is an edge / 0 otherwise (not the edge)}
let Pk-ij = # directed walks of length k(edges) from vi to vj, then A^k = {Pk-ij}
- 집합의 차수가 변의 수만큼 경로 점에서 점으로
ak-ij denote the (i, j)th entry in A^k
PF: by induction p(k) = "THM is true for k" = ∀ij ak-ij = pk-ij
Base case: k = 1, edge vi -> vj: Pij = 1 = aij(1)
Assume P(k), P(k+1)ij = ∑p(k)ij(_h:vk->vj is edged in G) = ∑p(k)ih*ahj = ∑a(k)jh*ahj = a(k+1)ij => matrix multiple, inductive step P(k)

DEF: a diagraph G = (V, E) is <u>strongly connected</u> if for all u, v ∈ V, ∃directed path from u to v in G
DEF: a directed graph is called a directed acyclic graph(DAG) if it does not contain any directed cycles

- Tournament graph

either u beats v : u -> v or v beats u : v -> u

ex) A -> B -> D -> E -> C (wait!) C -> A

ex) C -> B -> D -> E -> A so who is the best player?

DEF: a Directed Hamiltonian Path is a directed walk that visits every vertex exactly once

THM: every tournament graph actually contains such a directed Hami-path

PF: by induction, n = # of nodes // P(n) = “every tournament graph on n-nodes contains ‘DHP’”

Base case: n = 1

Inductive step: Assume P(n) is true // consider TPG on n+1 nodes

take out v -> this gives a TG on n-nodes

by P(n) v1 – v2 ... vn

case1: v -> v1

case2: v1 -> v smallest I such that v -> vi

v1 -> ... vi-1 -> vi -> ... vn

add v in here

v -> vi-1 // vi-1 -> v = are contradiction => so largest I such that vi -> v

- Chicken Tournament

either chicken v pecks chicken v : u -> v and vice wersa

u <u>virtually pecks</u> v if – u -> v or ∃w(another chicken) u -> w -> v

a chicken that virtually pecks every other chicken is called <u>King Chicken - check original graph</u>

THM: the chicken with highest out-degree is a king

PF: by contradiction, let u have highest out-degree // suppose u is not king

∃v v -> u and ∀w !u -> w(w -> u) or !w -> v(v -> w)

if u -> w then v – w

out-degree (v) >= out-degree(u) + 1 => contradiction