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

Abstract:
Graph theory is blueprint of pairing method. Through this theory we can organize matching the node with efficient and clever way. Futhermore, we expand this theory to reach to the algorithm or computer science for building system stable.

For the English:
ground, tackle, disparity, vertex(vertices, node), cardinality, bogus, equivalent, chromatic

Lec 6. GRAPH THEORY & COLORING:

This lecture use gender agenda to show how Graph theory works.

Chicago claims men have 74% more opposite gender partner than women.
ABC news claims // 233%

Graph is bunch of connected dots

DEF: a graph G is a pair of sets(V, E) where
V is nonempty set of items called vertices or nodes
E is a set of 2-items subsets of V called Edge (line)

DEF: two nodes xi and xj are adjacent if {xi, xj} is edge (linked)
DEF: an Edge e = {xi, xj} is incident to xi and xj
DEF: the # of edges incident to a node is the degree of the node ex) deg(x5) = 3
DEF: a graph is simple if it have no loops or multiple edges = we don't allow this

Vm = # of men, Vw is # of women, |V| is # of American
|V| /= 300M, |Vm| /= 147.6M, |Vw| /= 152.4M, |E| = ??
- we want to know the ratio of average degree of men and women

DEF: Am = average # of opposite gender partner of men and vice versa
what is Am/Aw ? => chicago = 1.74, ABC = 3.33

Am = ∑deg(x -< Vm) / |Vm| = |E|/|Vm| and vice versa
so Am/Aw is equal to |Vm|/|Vw| = 1.0325

- Graph coloring
given a graph G and K colors, assign a color to each nodoe so adjacent node get different colors
DEF: minimum value of K for which such a color exists is the chromatic number of G

+ if graph got triangle that means 2 color chromatic number are not possible
+ NP completeness : I got know 1 solution, that could apply it to all other problems

Basic coloring algorithm for G = (V, E)
1. order the nodes v1, v2 ,... vn
2. order the colors c1, c2 ... cn
3. for I = 1, 2 ... n // Assign the lowest legal(not redundant) color
Different numbering -> different # of color

THM: if every node im G has degree =< d(biggest degree in G), Basic Alg uses at most d+1 colors for G

PF: by induction -> P(d) = disaster // put n in 'node' or 'edge'
Base case: n = 1 => edges d =0 // 1 color = d+1
Inductive step: Assume P(n) is true for induction, Let G -(V, E) be any (n+1)-node graph, Let d = max degree in G
-Order the nodes v1, v2 ... vn, vn+1
-Remove vn+1 from G to create G' = (V', E')
G' has max degree =< d & n-nodes so P(n) says Basic Alg uses =< d+1 colors for v1 ~ vn

Vn+1 has =< d neighbors => for some color in {c1 ~ Cd+1} not used by any neighbor give Vn+1 that color => BasicAlg uses =< d+1 colors on G => P(n+1)

Kn = n-node complete graph = clique
d = n-1
X(kn) = n = d+1

DEF: a graph G = (V, E) is <u>bipartit(+ -> + shape)</u> if V can be split into Vl, Vr so that all edges connect a node in Vl to Vr