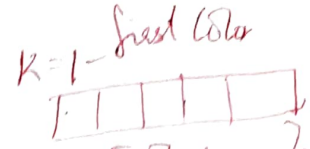
Graph Coloring.

m – no. of Colors.

Algorithm : mcoloring(k)

{



while (true)

{ NextValue(k); – which color to // assign x[k] to a legal color

if(x[k]=0) then // no such color possible.

return;

if (k=n) then

write (x[1:n]);

else

mcoloring(k+1);

mcoloring(2)

mcoloring(3)

}

The ~~below~~ above algm. finds legal colors for k^{th} vertex of graph. A value for x[k] is determined as the range(1..m). x[k] is assigned the next highest numbered color while maintaining distinct from the adjacent vertices of vertex k.

If no such color exists then x[k] is assigned. 0 x[1]... x[k-1] have been assigned integer values in the range [1,m] such that adjacent vertices have distinct integers

Graph is represented by its boolean adjacency matrix.

Algorithm NextVal(k)

{

while (true)

{

x[k] = (x[k]+1) mod (m+1); // next highest color.

if (x[k]=0) then

return; // all colors have been used.

for j=1 to n do // check if this color is distinct vertices

// from adjacent vertices colors.

{

if ((G[k,j]≠0) and x[k]=x[j]) then

break; // if (k,j) is an edge of adjacent

// vertices have same colors.

if (j=n+1) then

return

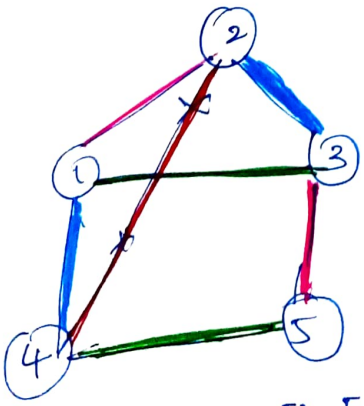
} // otherwise try to find another color.

m = no. of colors = 3

then x = 1, 2, 3

not more than 3.

Ex:



$$G = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \begin{array}{c} \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\ \begin{bmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix} \end{array}$$

x[1]	x[2]	x[3]	x[4]	x[5]
0	0	0	0	0
1	2			

K = 1

m = 3

R, B, G

(1, 2, 3)

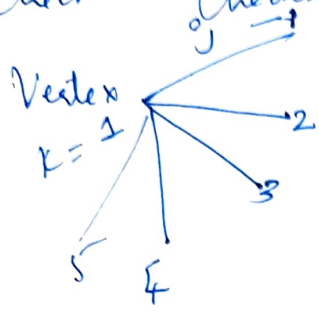
finding color for vertex 1

mcoloring (1) ⟶ NextValue [1] → x[1] = (0+1) mod (3+1)

= 1 mod 4

x[k] → x[1] = 1

Check. Checking adjacency. (refer matrix).



1 — 1 (Edge b/w 1 → 1 - no → value = 0)

1 — 2 → 1 (Check if both have same color?)

1 — 3 — 1

1 — 4 — 1

0 - value of color.

(x[k] = 1.

x[1] = 0

no color

Condition false)

∴ algorithm will send x[k] = 1.

Coz none of the color matches as initially they are set at value 0.

∴

x[1]	x[2]	x[3]	x[4]	x[5]
1	0	0	0	0
1	2			

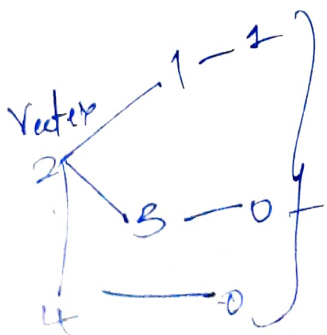
k = 2

Sending value k = 2 to mcoloring(2) algm

mcoloring (2) ⟶ NextValue (2)

x[2] = (0+1) mod 4 = 1

but 1 is already assigned to Vertex 1.



Color assigned to 3 will be 3. as 2 is not assigned to it or adjacent ones of 1 & 3 by this way algm will run.