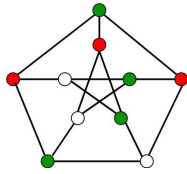


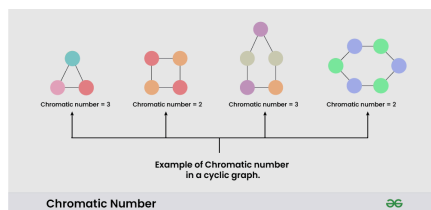
COLORING OF A GRAPH

Graph coloring refers to the problem of **coloring vertices** of a graph in such a way that **no two adjacent** vertices have the **same color**. This is also called the **vertex coloring** problem. If coloring is done using at most m colors, it is called m -coloring.



CHROMATIC NUMBER

The **minimum** number of **colors** needed to **color** a graph is called its **chromatic** number. For example, the following can be colored a minimum of 2 colors.



Algorithm steps for coloring of a graph:

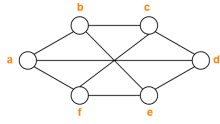
1. Assign colors one by one to different vertices, starting from vertex 0.
2. Before assigning a color, check if the adjacent vertices have the same color or not.
3. If there is any color assignment that does not violate the conditions, mark the color assignment as part of the solution.
4. If no assignment of color is possible then backtrack and return false.

Applications of Graph Colouring

- Design a timetable.
- Sudoku.
- Register allocation in the compiler.
- Map coloring.
- Mobile radio frequency assignment

Problem-01:

Find chromatic number of the following graph-



Solution-

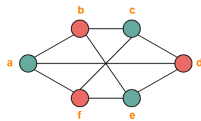
Applying Greedy Algorithm, we have-

Vertex	a	b	c	d	e	f
Color	C1	C2	C1	C2	C1	C2

From here,

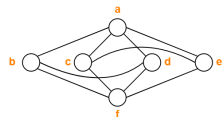
- Minimum number of colors used to color the given graph are 2.
- Therefore, Chromatic Number of the given graph = 2.

The given graph may be properly colored using 2 colors as shown below-



Problem-02:

Find chromatic number of the following graph-



Solution-

Applying Greedy Algorithm, we have-

Vertex	a	b	c	d	e	f
Color	C1	C2	C2	C3	C3	C1

From here,

- Minimum number of colors used to color the given graph are 3.
- Therefore, Chromatic Number of the given graph = 3.

The given graph may be properly colored using 3 colors as shown below-

