Chapter 5
Backtracking

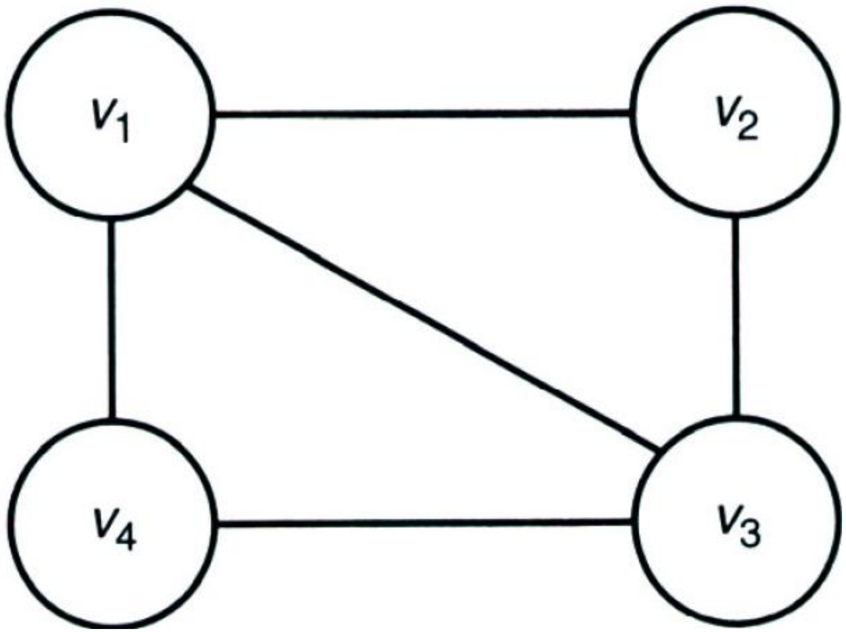
Graph coloring

- The *m*-Coloring problem
  - Finding all ways to color an undirected graph using at most *m* different colors, so that no two adjacent vertices are the same color.
  - Usually the *m*-Coloring problem consider as a unique problem for each value of *m.*

Example

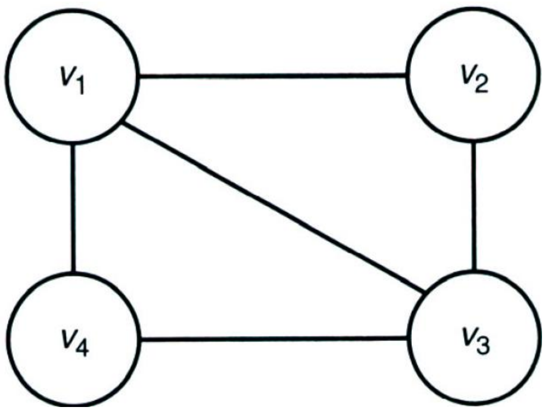
- 2-coloring problem
  - No solution!
- 3-coloring problem

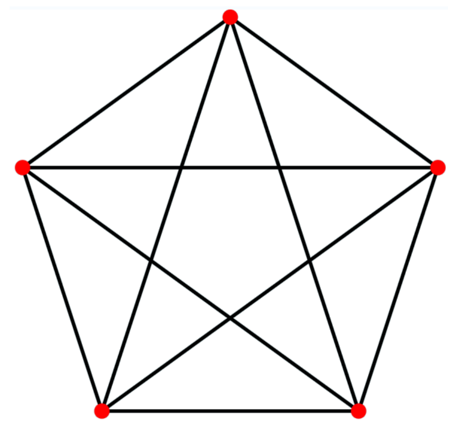
Vertex	*Color*
*v*1	color1
*v*2	color2
*v*3	color3
*v*4	color2



Application: Coloring of maps

- **Planar** graph
  - It can be drawn in a plane in such a way that no two edges cross each other.

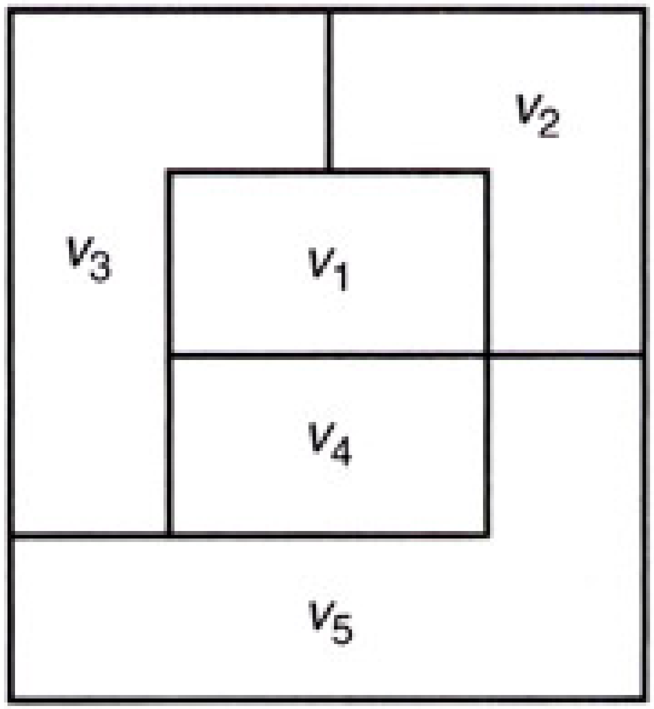




- To every map there corresponds a planar graph

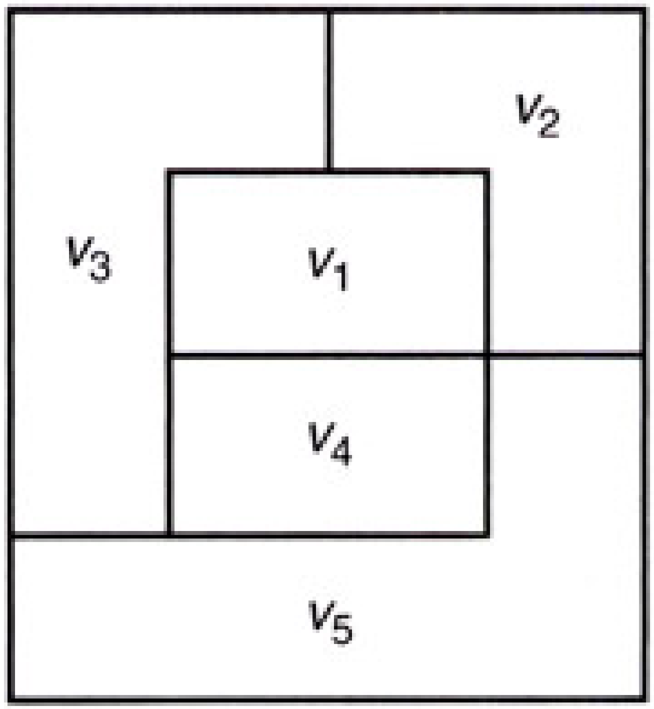
Example (1)

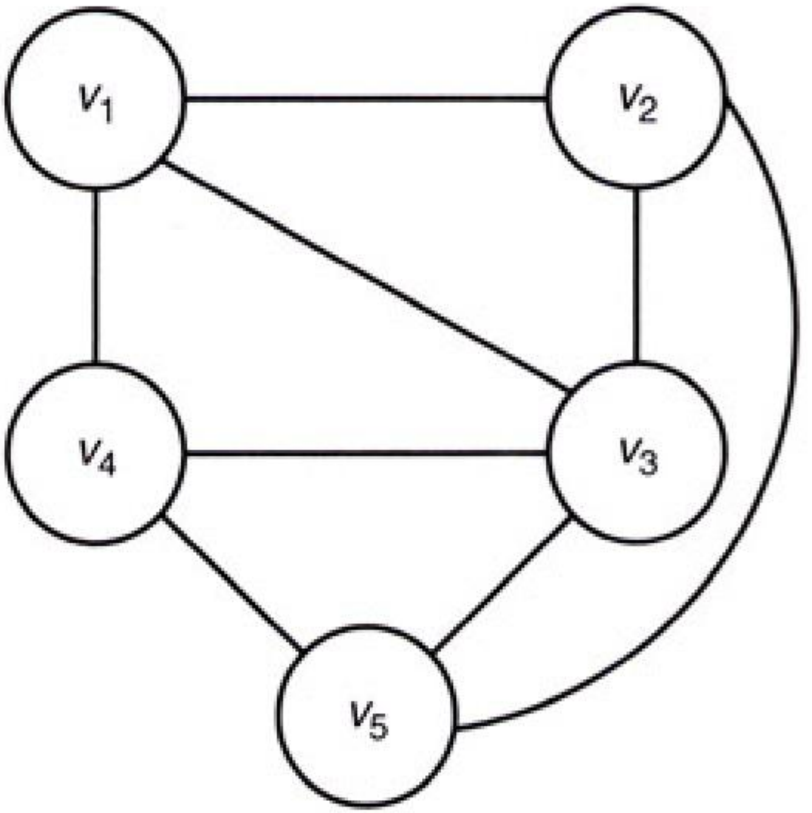
- Map



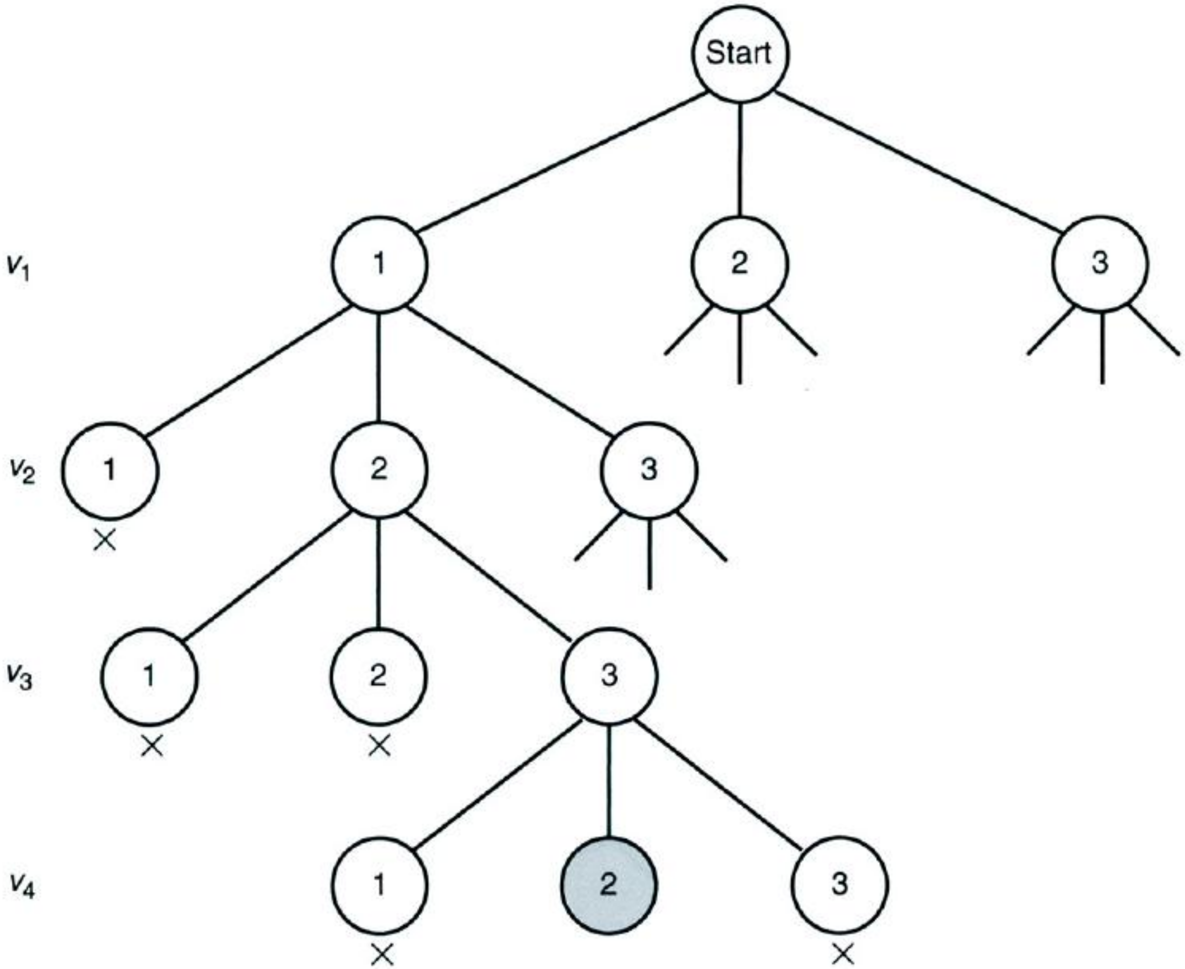
Example (2)

- corresponded planar graph





The pruned state space tree



Algorithm for Graph Coloring

```
mColoring (k)
{
Repeat{
Next_Value(k)
If(x[k]=0) then return
if/(k=n) then
Write(x[1:n]);
Else
mColoring(k+1);
}until(false);
}
```

Algorithm for nextvalue

```
Next_Value(k)
{
Repeat
{
X[k]=x[k]+1 mod(m+1)
If(x[k]=0) then return;
For j= 1 to n do{
If(( G[k,j]≠0) and(x[k]=x[j]))
Then break;
}
If(j=n+1) then return;
}until(false);
}
```

- The top level call to *m*_coloring
  - m_coloring(0)
- The number of nodes in the state space tree for this algorithm

$$1 + m + m^2 + \cdots + m^n = \frac{m^{n+1} - 1}{m - 1}$$

Assignment

- Q.1)What is Backtracking?
- Q.2)What is application of Graph coloring?
- Q.3)Explain graph coloring with example.