a) Say the 2 colors are black and white.
   For each vertex in G,
   if u is not colored
   Color [u] ← black ← for each start vertex of disjointed subgraph
   Q ← {u}
   While Q is not Empty
     t ← Dequeue [Q]
     for each v ∈ adj [t]
     if v is not Colored
       Color [v] = ~ Color [t]
       Enqueue (Q, v)

   Complexity = O(V+E)

b) Decision Problem.
   For an undirected graph G(v,e) and a positive integer
   k ≥ 2, is there a k-coloring for G?

d) For any truth assignment t for ϕ.
   Set C(xi) = ϕ(xi) C(¬xi) = ϕ (¬xi) i = 1...n
   Then one of xi and ¬xi is colored c(true) and the other c(false)

Hence the graph G containing only the literal edges is
with each ending at 'RED'

This function C is a 3-coloring of G.
e. Take the case \( \pi = \psi = \Phi = \text{false} \) and show that it will not give a 3 colorable graph.

\[
\begin{array}{c}
\text{false} \\
\text{false} \\
\text{false} \\
\text{true}
\end{array}
\]

Since \( \pi = \psi, \text{false}, \) 3 must be false. Leaving 1, 2 with assignment true or false, then we have both 2 and 3 with false. But now we have to assign a 'true' vertex with false - contradiction.

Since \( \pi, \psi, \Phi \) can only be assigned true or false, then one of them has to be assigned true.

f) 3 Color is NP. Given a graph we can verify it in polynomial time. Whether \( C(u) \neq C(v) \) for every edge \( (u,v) \in E \).

To show 3SAT \( \equiv \) 3 Color

Transformation as in the textbook. For any clause \( C_i \) construct a widget with 3 literals in the clause.

Proof of Equivalence:

If there is a truth assignment for \( \phi \) then every \( C(i) = \text{true} \). It means at least one literal in \( C_i \) is true.

Color the 3 special vertices as \( C(\text{true}) \), \( C(\text{false}) \) and \( C(\text{red}) \).

Color every vertex of variables and negation with

\[
C(x_i) = C(+x_i) \quad C(-x_i) = C(+(-x_i))
\]

For any widget, at least one literal is colored \( C(\text{true}) \).
According to (c) it is 3-colorable. It means every clause edge satisfies 3-coloring and color 'clause' edge satisfies 3-coloring. According to (d) every literal edge satisfies 3-coloring.

So C is a 3-coloring of G.

Now we know \(3\text{-SAT} \leq \text{3-coloring}\)

and \(3\text{-SAT} \leq \text{NPC}\).

\(\therefore 3\text{-Coloring} \leq \text{NPC}\).

Q2: a) 

![Graph image]

Dominating set \(\{a, c\}\)

b) Decision Problem:

Does there exist a subset \(V' \subseteq V\) such that for \(|V'| \leq k\) for every vertex in \((V - V')\) is linked to at least one vertex in \(V'\) such that \(e \in E\)?

Proof

We choose Vertex Cover (VC) as the source problem and will show that it is a reducible to our dominating set problem (DSP). First note that DSP \(\in \text{NP}\), since a non-deterministic algorithm need to find only a subset \(V'\) of \(V\) and check in polynomial time that there exists an edge between \((V - V')\) and \(V\) in the original graph.

Now let's transform VC to DSP.
Let say $V'$ is a VC for a graph $G(V,E)$.
As per the hint we can add additional vertices to the set $V'$ and it will still remain a VC.

Now $<V_1,V_2,...,V_k>$ where $|V'| = \text{size of VC}$.

edges $<e_1,e_2,...>$ total edges on the graph.

Each edge $e_i$ on VC must be linked with at least one member in $V'$.

Now for every edge $(u,v)$ add 2 edges $(u,x)$ and $(x,v)$.

2 cases to analyse:

Case 1: If both $u,v$ are members of VC, then additional vertex $x$, does not affect the VC.

Case 2: If only one of $(u,v)$ is in VC, in order to cover edge $(u,x)$ $(x,v)$ we have to choose at least one more from $u,v,x$.

Say we choose $x$.

Now after selecting all such vertices, members of $(V-V')$ will have at least one edge between them and the $V'$ set, since it's a VC. But this has however now reduced to a dominating set problem. Reduction complete. $\text{VC} \leq \text{DSP}$.

But we know that VC is NPC

$\Rightarrow \therefore \text{DSP is NPC.}$