Q1 Solution:

1. **General ILP for minimum vertex cover:**
    a. Introduce a decision variable x_i, $\forall$ i $\in$ V (vertices)
        i. $x_i = 1$ => we put i in the cover
        ii. $x_i = 0$ => we don't put i in cover
    b. We then formulate the weighted vertex cover using cost functions and linear constraints on the variables x_i as C = $(c_1, c_2, c_3, \ldots, c_v)$
    c. Minimize total weight of vertices in V subject to vertices in C form a cover of total weight = $\Sigma_{v \in C} C_v$
    d. Instead of taking the weight for every vertex we take the formulated variable weight multiplied by vertex x_i (if it is in x_i we take $x_i = 1$ and 0 otherwise)
    e. Minimize $\Sigma_{v \in V} C_v . x_v$, subject to the vertices form a cover if
        i. $u \in \text{Cover}$ or $v \in \text{Cover}$ $\forall (u, v) \in E$
        ii. Implies that $x_u = 1$ or $x_v = 1$, $\forall (u, v) \in E$
        iii. $x_u + x_v >= 1, \forall (u, v) \in E$
        iv. $x_i \in \{0, 1\}, \forall i \in V$

Algorithm:

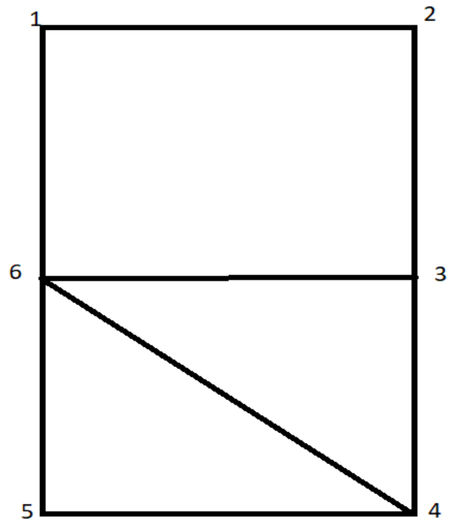
Begin: Weighted_Vertex_Cover_LP(G)

1. Generate LP corresponding to G
    - Minimize $\Sigma_{v \in V} C_v . x_v$
    - Subject to $x_u + x_v >= 1, \forall (u, v) \in E$
    - $x_i \in \{0, 1\}, \forall i \in V$ -> this is not a linear constraint and that's why it is called as integer linear programming
2. Solve the linear programming problem
3. Cover ← $\{v \in V: x_v = 1\}$
4. return Cover

End

Note: This is not a Linear programming problem, it is a 0/1 Linear problem. Unfortunately, 0/1 Linear problem is Np-hard (Solution isn't feasible)

Given graph:



1. Objective is to Minimize $1 . x_1 + 2 . x_2 + 3 . x_3 + 4 . x_4 + 5 . x_5 + 6 . x_6$

   Subject to $x_1 + x_2 >= 1$, $x_1 + x_6 >= 1$, $x_2 + x_3 >= 1$, $x_3 + x_6 >= 1$, $x_4 + x_6 >= 1$, $x_5 + x_6 >= 1$, $x_4 + x_5 >= 1$, $x_3 + x_4 >= 1$

   We must add 0/1 constraints for the variables to state that the values come from 0 or 1. So, $x_i \in \{0, 1\}, \forall\, i \in V$ is the additional constraint

   If we represent this as the matrix form, we get AX >= B where,

   $$A = \begin{pmatrix} 1\,1\,0\,0\,0\,0 \\ 0\,1\,1\,0\,0\,0 \\ 0\,0\,1\,1\,0\,0 \\ 0\,0\,1\,0\,0\,1 \\ 0\,0\,0\,1\,0\,1 \\ 0\,0\,0\,1\,1\,0 \\ 1\,0\,0\,0\,0\,1 \\ 0\,0\,0\,0\,1\,1 \end{pmatrix} \quad X = \begin{pmatrix} x1 \\ x2 \\ x3 \\ x4 \\ x5 \\ x6 \end{pmatrix} \quad B = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

   Solving the vertex cover problem, we get the optimal minimum vertex cover

   set as {2, 4, 6}

   thus, we have the integer constraints as $x_2 = x_4 = x_6 = 1$

   and $x_1 = x_3 = x_5 = 0$

Considering the greedy approach (starting with the vertex having minimal weight), we get $x_1 = x_3 = x_5 = x_4 = 1$ and $x_2 = x_6 = 0$ so, the sub-optimal cost is 13 whereas optimal $x_2 = x_4 = x_6 = 1$ and $x_1 = x_3 = x_5 = 0$ cost is 12

Sub-optimal value <= 2 * optimal value 13 <= 2*12, 13 <= 24

<u>Q2 Solution:</u>

1. **General ILP for Minimum Dominating Set**
   - **a.** Introduce a decision variable x_i, $\forall$ i $\in$ V (vertices)
     - **i.** $x_i = 1$ => we put i in the cover
     - **ii.** $x_i = 0$ => we don't put i in cover
   - **b.** We then formulate the weighted vertex cover using cost functions and linear constraints on the variables x_i as $D = (d_1, d_2, d_3, \ldots, d_v)$
   - **c.** Minimize total weight of vertices in V subject to vertices in D form a dominant set of total weight = $\Sigma_{v \in D} d_v$
   - **d.** Instead of taking the weight for every vertex we take the formulated variable weight multiplied by vertex x_i (if it is in x_i we take $x_i = 1$ and 0 otherwise)
   - **e.** Minimize $\Sigma_{v \in D} d_v \,.\, x_v$, subject to the vertices form a cover if
     - **i.** $u \in$ Dominant Set or all the adjacent vertices in Dominant set , $\forall (u, v) \in E$
     - **ii.** Implies that $x_u = 1$, $x_u \in$ Dominant Set, $x_i + \Sigma_{j \in N(i)} x_j \geq 1$, N = {Neighboring vertices of i}
     - **iii.** $x_i + \Sigma_{j \in N(i)} x_j \geq 1$, $\forall$ neighboring edges $\in$ E $\rightarrow$ this condition is to ensure the domination
   - **f.** $x_i \in \{0, 1\}$, $\forall$ i $\in$ V

Algorithm:

Begin: Weighted_Dominating_Set_LP(G)

1. while there exist, non-dominated vertices do

   randomly select a vertex v from all non-dominated vertices;

   select a vertex u ∈ N[v] with the highest degree.

   D := D ∪ {u};

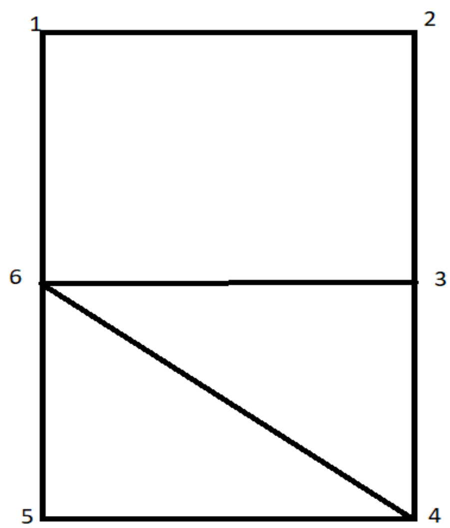
2. Generate LP corresponding to G

- Minimize $\Sigma_{v \in D} d_v \cdot x_v$
- Subject to $x_i + \Sigma_{j \in N(i)} x_j \geq 1$, N = {Neighboring vertices of i}
- $x_i \in \{0, 1\}$, $\forall$ i $\in$ V -> this is not a linear constraint and that's why it is called as integer linear programming

3. Solve the linear programming problem
4. Dominating Set ← $\{v \in V: x_v = 1\}$
5. return Dominating Set

End

Given graph:



Objective: We want to minimize the number of vertices in the dominating set. So, the objective is to minimize $1 \cdot x_1 + 2 \cdot x_2 + 3 \cdot x_3 + 4 \cdot x_4 + 5 \cdot x_5 + 6 \cdot x_6$

Each vertex needs to be in the set or have a neighbor in the set. So, we get a constraint for each vertex of the form $\Sigma_{i \in N(v)} x_i \geq 1$, for all $v \in V$.

Subject to

$x_1 + x_2 + x_6 >= 1$, $x_1 + x_2 + x_3 >= 1$, $x_2 + x_3 + x_4 + x_6 >= 1$, $x_3 + x_4 + x_5 + x_6 >= 1$, $x_4 + x_5 + x_6 >= 1$, $x_1 + x_3 + x_4 + x_5 + x_6 >= 1$

$x_i \in \{0, 1\}$

if we represent this as a matrix form, we get AX >= B

A = $\begin{pmatrix} 1\ 1\ 0\ 0\ 0\ 1 \\ 1\ 1\ 1\ 0\ 0\ 0 \\ 0\ 1\ 1\ 1\ 0\ 1 \\ 0\ 0\ 1\ 1\ 1\ 1 \\ 0\ 0\ 0\ 1\ 1\ 1 \\ 1\ 0\ 1\ 1\ 1\ 1 \end{pmatrix}$ X = $\begin{pmatrix} x1 \\ x2 \\ x3 \\ x4 \\ x5 \\ x6 \end{pmatrix}$ B = $\begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$

Solving the Dominating set problem, we get {1, 6}

thus, we have the integer constraints as $x_1 = x_6 = 1$ and $x_2 = x_3 = x_4 = x_5 = 0$

Considering the greedy approach (starting with the vertex having minimal weight), we get $x_1 = x_4 = 1$ and $x_2 = x_3 = x_5 = x_6 = 0$ so, the optimal cost is 5.