Minimum Vertex Cover Problem:

1) fractional relaxation for an arbitrary graph G= (V, E)

   Answer:

   - Given a graph G = (V, E) and the vertex weights = Weights(V)
     Minimum vertex cover problem for graph G as an ILP by using a variable x_v for each vertex
     vertex v, taking 0 or 1 values.
   - $x_v = 0$ means $v \notin$ Cover C, $x_v = 1$ means $v \in$ Cover C
   - The weight of the solution, which we want to minimize, is $\Sigma_{v \in V} x_v * W(v)$ such that $x_u + x_v >= 1$ for each edge (u, v)
   - This gives the Integer Linear Programming as
     Minimize $\Sigma_{v \in V} x_v * W(v)$
     Subject to $x_u + x_v >= 1$ for all $(u, v) \in$ Edges E
     $X_v <= 1$ for all $v \in$ Vertices V
     $X_v \in N$ for all $v \in$ Vertices V
   - Relaxing Integer Linear Programming to a linear program
     Minimize $\Sigma_{v \in V} x_v * W(v)$
     Subject to $x_u + x_v >= 1$ for all $(u, v) \in$ Edges E
     $X_v <= 1$ for all $v \in$ Vertices V
     $X_v >= 0$ for all $v \in$ Vertices V
   - Linear program can be solved in polynomial time
   - Suppose x* is an optimal solution to the linear program, rounding each value to the closest integer i.e.,
     $x'_v = 1$ if $x^*_v >= ½$
     $x'_v = 0$ if $x^*_v < ½$
   - The set of valid vertex cover C, because of each edge (u, v) has the conditions
     $x^*_u + x^*_v >= 1$
     so at least one of x^*_u or x^*_v must be at least ½ and at least one of u or v belongs to Cover C
   - The cost of cover is at most twice the optimum

2) Dual linear program for an arbitrary graph G= (V, E)

   Answer:

   - For the dual, the number of variables is equal to the number of vertices in the minimum vertex cover
   - Using weak duality, we convert minimum vertex cover to maximum matching in a bipartite graph
   - As we know, the LP for minimum vertex cover is

   Minimize $\Sigma v \in V\ xv*W(v)$

   Subject to $xu + xv >= 1$ for all $(u, v) \in$ Edges E
   $Xv >= 0$ for all $v \in$ Vertices V

   - Its dual has one variable y(u,v) for edge (u, v) and it is

   Maximize $\Sigma\ (u, v) \in V\ y(u,v)$

   Subject to $\Sigma\ u{:}\ (u, v) \in V\ y(u,v) <=$ Weights(v) for all $v \in$ Vertices V
   $y(u,v) >= 0$ for all $(u, v) \in$ Edges E

   - Assigning a non-negative charge to each edge such that the total charge
   Overall the edges are as large as possible.
   - The sum of charges is a lower bound to the weight of the minimum vertex cover
   In the weighted graph, G = (V, E) with weights W

3) ILP corresponding to the dual LP (Maximum matching)

Answer:

- Using weak duality, we convert minimum vertex cover to maximum matching in a bipartite graph
- The input is a graph with each edge having a positive weight W_{uv}
- A maximum weighted bipartite matching is defined as a perfect matching where the sum of values of the edges in the matching have a maximum value.
- The size of such a matching is n^2.
- graph is not complete bipartite then missing edges are inserted with the value zero
- The Variable x_{uv} is defined as,
  $x_{uv = 0}$ if the edge (u, v) belongs to the matching
  $x_{uv = 1}$ if the edge (u, v) does not belong to the matching
- ILP for this graph is
  Maximum $\Sigma W_{uv}\, x_{uv}$
  For all u, $\Sigma x_{uw} = 1$
  For all v, $\Sigma x_{pv} = 1$
  For all u, v $x_{uv >= 0}$ is integral

4) write in words the graph problem corresponding the dual ILP

- A matching in a Bipartite Graph is a set of the edges chosen in such a way that no two edges share an endpoint.
- A maximum matching is a matching of maximum size (maximum number of edges).
- In a maximum matching, if any edge is added to it, it is no longer matching.
- There can be more than one maximum matchings for a given Bipartite Graph.
- Given a bipartite graph G = (A ∪ B, E), find an S ⊆ A × B that is a matching and is as large as possible.
- A matching *M* is maximal if and only if there does not exist an augmenting path with respect to *M*.
- Algorithm

bipartiteMatch (u, visited, assign)

Input: Starting node, visited list to keep track, assign the list to assign node with another node.

Output: Returns true when a matching for vertex u is possible.

```
Begin
  for all vertex v, which are adjacent with u, do
     if v is not visited, then
        mark v as visited
        if v is not assigned, or bipartiteMatch(assign[v], visited, assign) is true, then
          assign[v] := u
          return true
  done
  return false
End
```

maxMatch (graph)

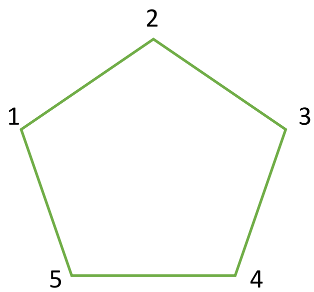
Input: The given graph.

Output: The maximum number of the match.

```
Begin
  initially no vertex is assigned
  count := 0
  for all applicant u in M, do
    make all node as unvisited
    if bipartiteMatch(u, visited, assign), then
      increase count by 1
  done
End
```

5) Given graph with 5 vertices and edges 1-2, 2-3, 3-4, 4-5, 5-1

a) ILP for Minimum Weighted Vertex Cover



1) Objective: Minimize $\Sigma_{a \in V}$ weight (V) * x_a
   Assuming weights of all vertices are equal to 1
   $x_1 + x_2 + x_3 + x_4 + x_5$

   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

2) Subject to:

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

   Such that $x_1 = x_3 = x_4 = 1$
   $X_2 = x_5 = 0$

the above can be written in matrix form as where variables = vertices = columns in A and constraints = edges = rows in A

$AX >= b$

we get $x_1 = x_2 = x_4 = 1$ and $x_3 = x_5 = 0$ solution is 3

ILP for Maximum Matching

1) Objective: Maximize $\Sigma_{(u, v) \in V}\, y(u,v)$
   $y_1 + y_2 + y_3 + y_4 + y_5 = y^T B$

   Subject to $\Sigma_{u: (u, v) \in V}\, y(u,v) <=$ Weights(v) for all $v \in$ Vertices V
   $y(u,v) >= 0$ for all $(u, v) \in$ Edges E

   y_1 $x_1 + x_2 <= 1$
   y_2 $x_2 + x_3 <= 1$
   y_3 $x_3 + x_4 <= 1$
   y_4 $x_4 + x_5 <= 1$

$y_5 \quad x_1 + x_5 <= 1$

$x_1 * (y_1 + y_5) + x_2 * (y_1 + y_2) + x_3 * (y_2 + y_3) + x_4 * (y_3 + y_4) + x_5 * (y_4 + y_5)$ is Maximized

$y_1 + y_5 >= 1$
$y_1 + y_2 >= 1$
$y_2 + y_3 >= 1$
$y_3 + y_4 >= 1$
$y_4 + y_5 >= 1$

$y_2 = y_3 = 1$ and $y_1 = y_4 = y_5 = 0$ The matching set is ([1,2], [3,4]) Assuming all edge weights are equal to 1 we get the solution as 2

b) Optimal Solution for Vertex Cover

Considering the greedy approach, we get $x_1 = x_2 = x_4 = 1$ and $x_3 = x_5 = 0$ so the sub-optimal cost is 3 whereas optimal $x_1 = x_2 = x_4 = 1$ and $x_3 = x_5 = 0$ cost is 3

Greedy approach <= 2 * optimal value
3 <= 2*3
3 <= 6

Optimal Solution for Maximum matching

Considering the greedy approach here too, $y_2 = y_3 = 1$ and $y_1 = y_4 = y_5 = 0$ The matching set is ([1,2], [3,4]) Assuming all edge weights are 1 we get 2

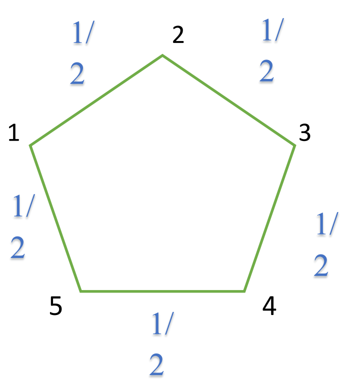
Greedy approach <= 2 * optimal value
2 <= 2*2
2 <= 4

c) Fractional Optimal solution for Minimum Vertex Cover

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

are the integral constraints

$2x_1 + 2x_2 + 2x_4 + 2x_3 + 2x_5 >= 5$
$x_1 + x_2 + x_4 + x_3 + x_5 >= 5/2$



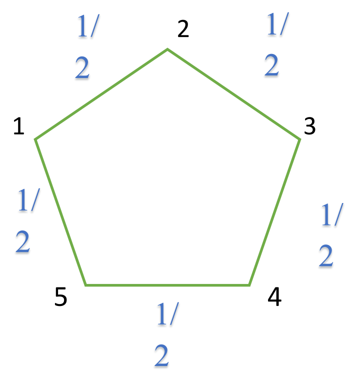
$x_1 = 0.5, x_2 = 0.5, x_3 = 0.5, x_4 = 0.5, x_5 = 0.5$

Fractional Optimal solution for Maximum Matching

$y_1 + y_5 >= 1$
$y_1 + y_2 >= 1$
$y_2 + y_3 >= 1$
$y_3 + y_4 >= 1$
$y_4 + y_5 >= 1$

are the integral constraints

$2y_1 + 2y_2 + 2y_4 + 2y_3 + 2y_5 >= 5$
$y_1 + y_2 + y_4 + y_3 + y_5 >= 5/2$



$y_1 = 0.5$, $y_2 = 0.5$, $y_3 = 0.5$, $y_4 = 0.5$, $y_5 = 0.5$

Here, The fractional solutions for minimum vertex cover and its dual (maximum matching) are equal.