Graph Theory

Dr. Irfan Yousuf
Department of Computer Science (New Campus)
UET, Lahore

(Lecture # 25; April 26, 2023)

Outline

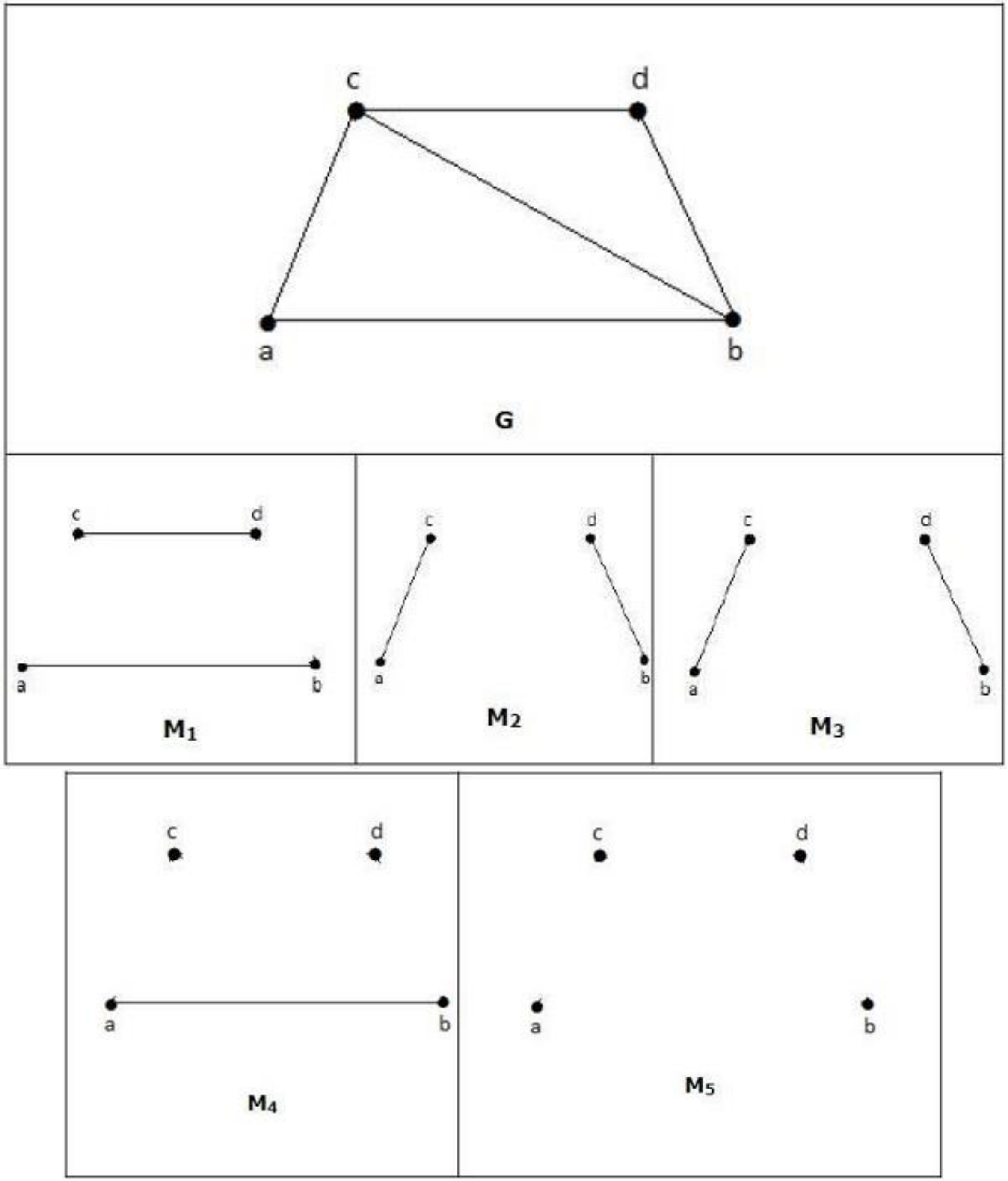
- Matchings in Graphs

Matchings in Graphs

- In graph theory, a matching in a graph is a set of edges that do not have a set of common vertices.
- In other words, a matching is a graph where each node has either zero or one edge incident to it.
- A matching graph is a subgraph of a graph where there are no edges adjacent to each other.
- Simply, there should not be any common vertex between any two edges.

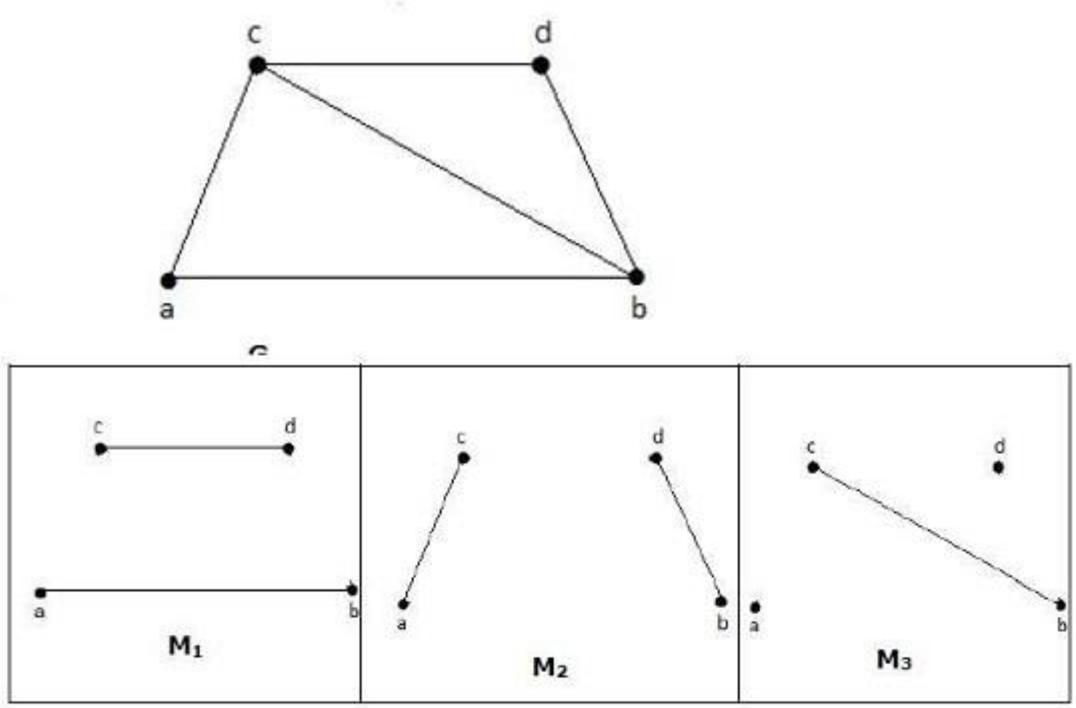
Matchings in Graphs

- Let G = (V, E) be a graph. A subgraph is called a matching M(G), if each vertex of G is incident with at most one edge in M.
  - $\deg(V) \leq 1 \ \forall \ V \in G$
- which means in the matching graph M(G), the vertices should have a degree of 1 or 0, where the edges should be incident from the graph G.


c
d
a
b
G
M_1
M_2
M_3
M_4
M_5

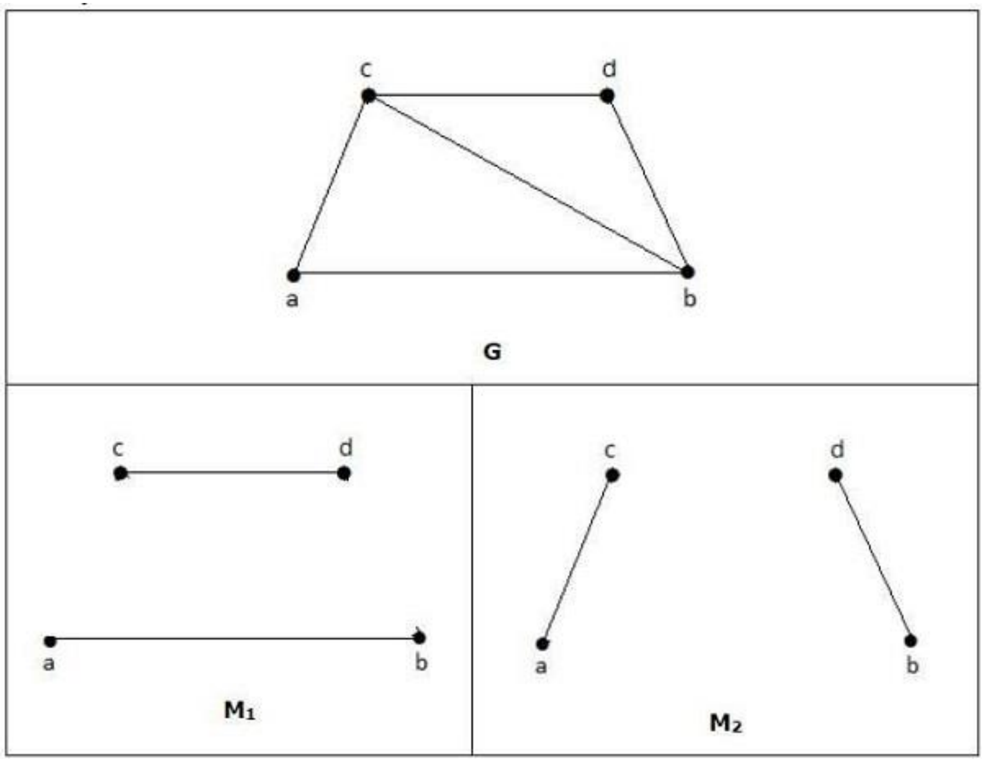
Maximal Matchings

- A matching M of graph ‘G’ is said to maximal if no other edges of ‘G’ can be added to M.

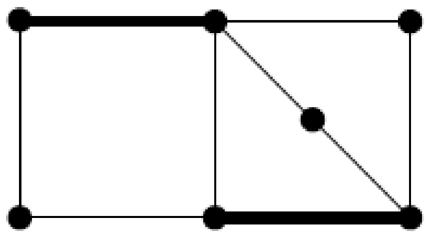


Maximum Matchings

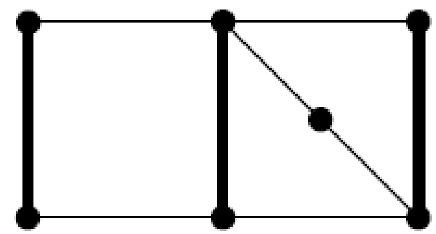
- It is also known as largest maximal matching. Maximum matching is defined as the maximal matching with maximum number of edges.
- The number of edges in the maximum matching of ‘G’ is called its matching number.



Maximal vs. Maximum Matchings



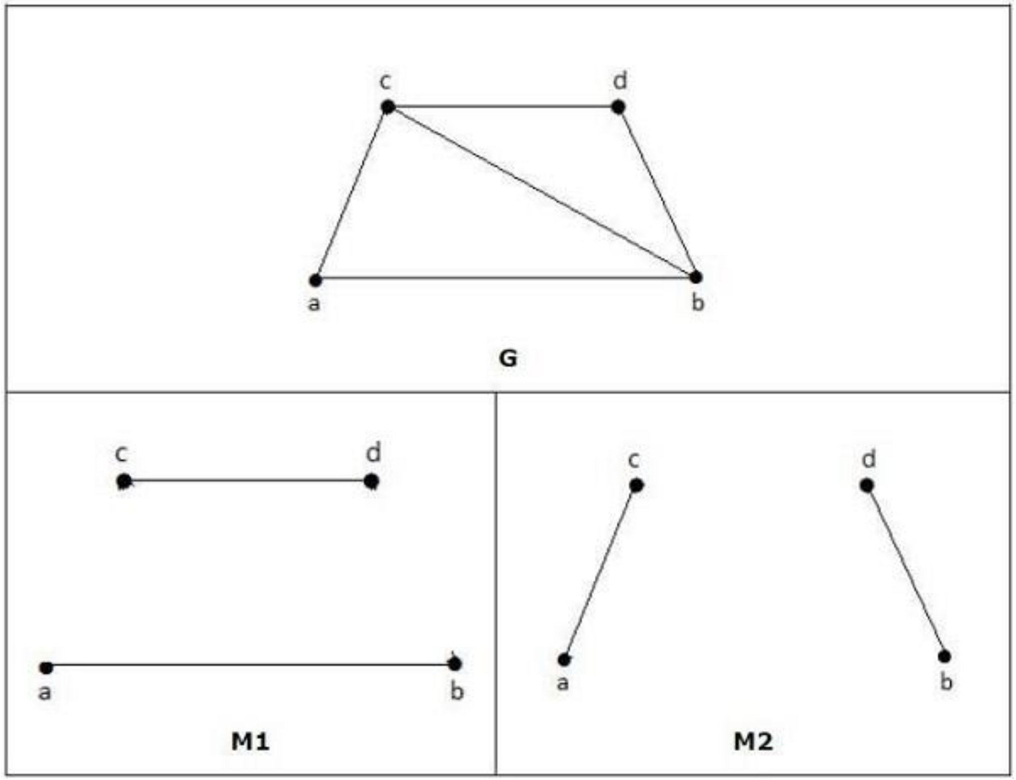
(a) *maximal* matching



(b) *maximum* matching

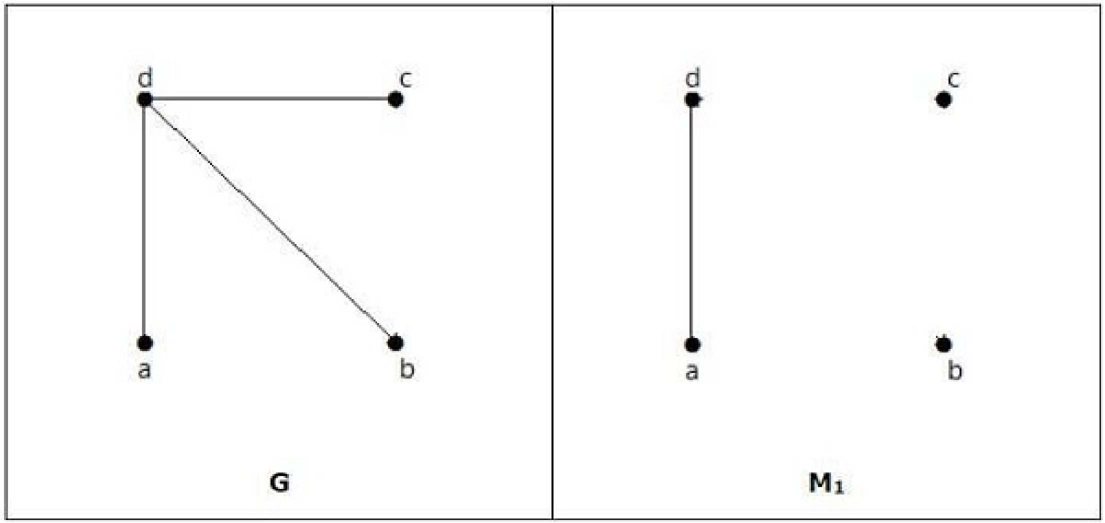
Perfect Matchings

- A matching (M) of graph (G) is said to be a perfect match, if every vertex of graph g (G) is incident to exactly one edge of the matching (M), i.e.,
- deg(V) = 1 ∀ V



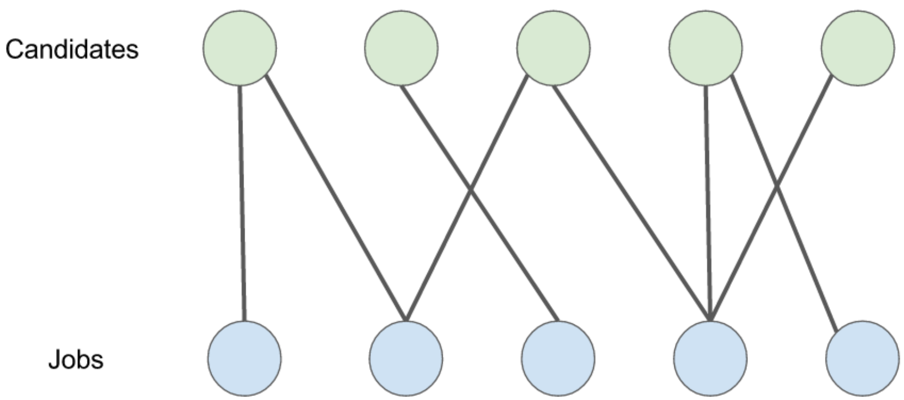
Perfect Matchings

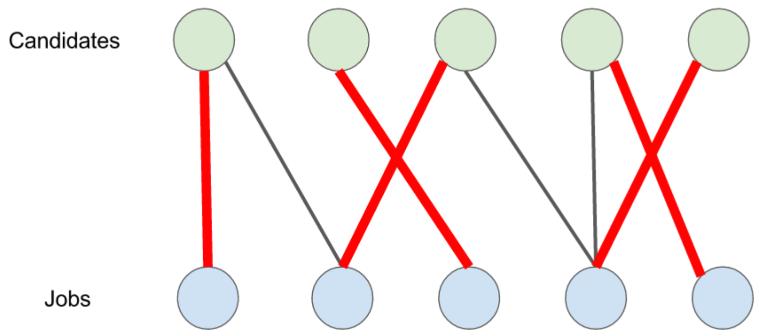
- Every perfect matching of graph is also a maximum matching of graph, because there is no chance of adding one more edge in a perfect matching graph.
- If a graph 'G' has a perfect match, then the number of vertices |V(G)| is even.
- The converse of the above statement need not be true. If G has even number of vertices, then M1 need not be perfect.



Applications of Matchings

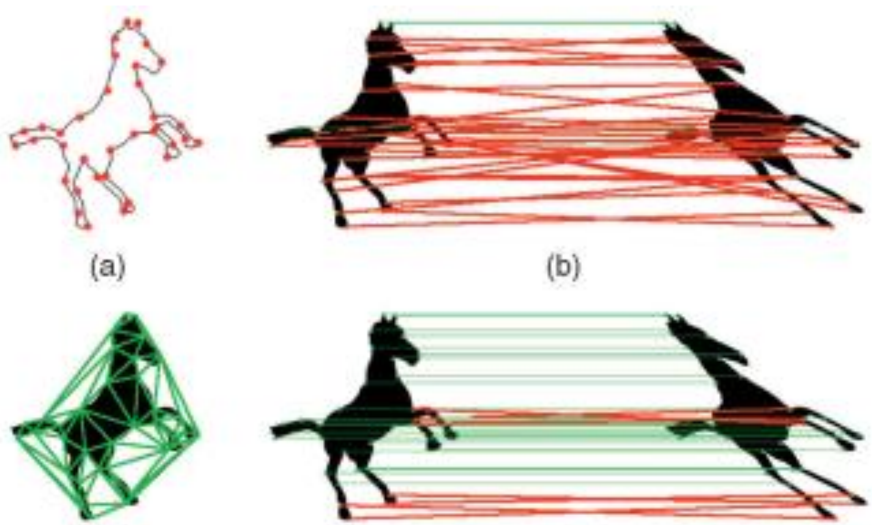
Say there is a group of candidates and a set of jobs, and each candidate is qualified for at least one of the jobs. We can use graph matching to see if there is a way we can give each candidate a job they are qualified for.

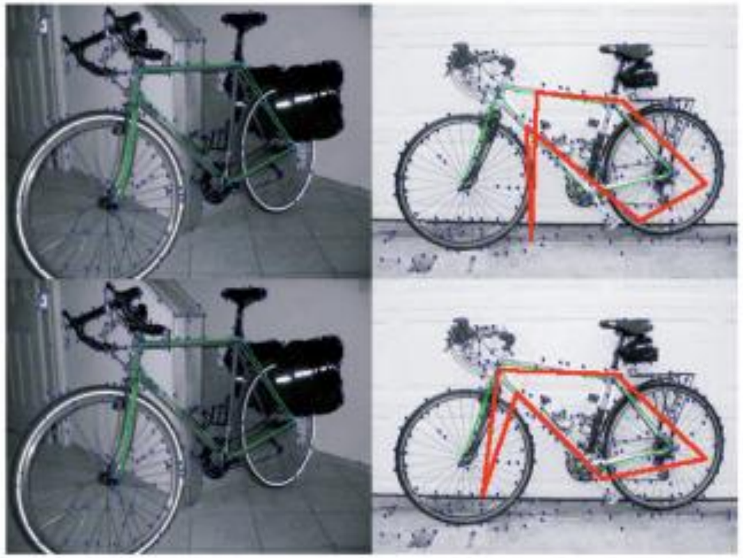




Applications of Matchings

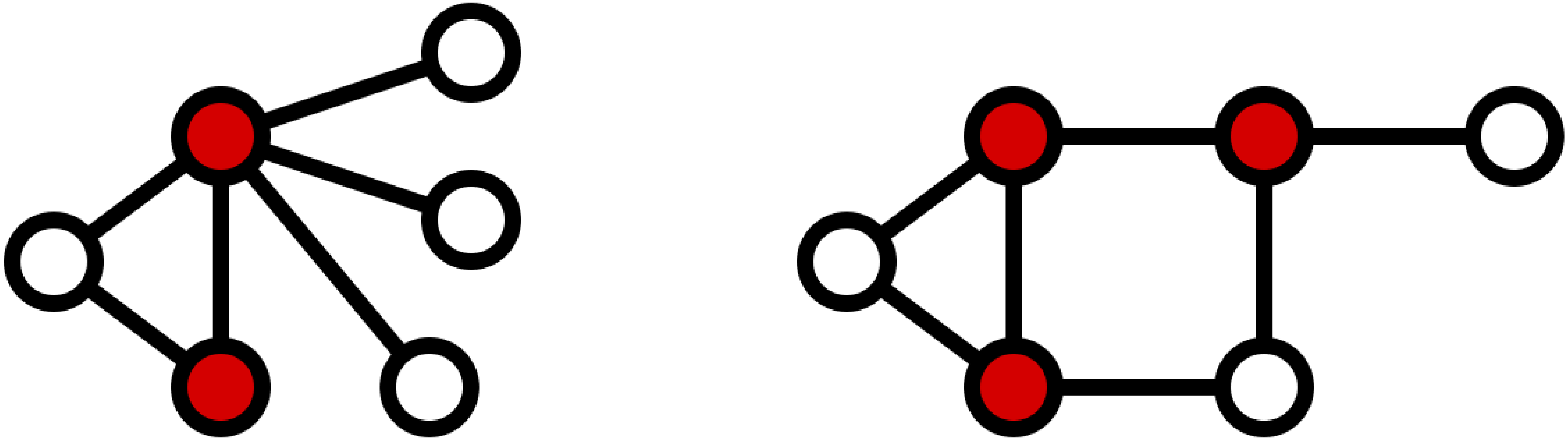
Pattern recognition is the ability of machines to identify patterns in data, and then use those patterns to make decisions or predictions using computer algorithms.





Vertex Cover

In graph theory, a vertex cover (sometimes node cover) of a graph is a set of vertices that includes at least one endpoint of every edge of the graph.



Vertex Cover

A **vertex cover** of an undirected graph $G = (V, E)$ is a subset $V' \in V$ such that if (u, v) is an edge of G, then either $u \in V'$ or $v \in V'$(or both).

Approximation Algorithm

- An algorithm that returns near-optimal solutions is called an approximation algorithm.
- An optimization problem in which each potential solution has a positive cost, and we wish to find a near-optimal solution.
- Depending on the problem, we may define an optimal solution as one with maximum possible cost or one with minimum possible cost; that is, the problem may be either a maximization or a minimization problem.

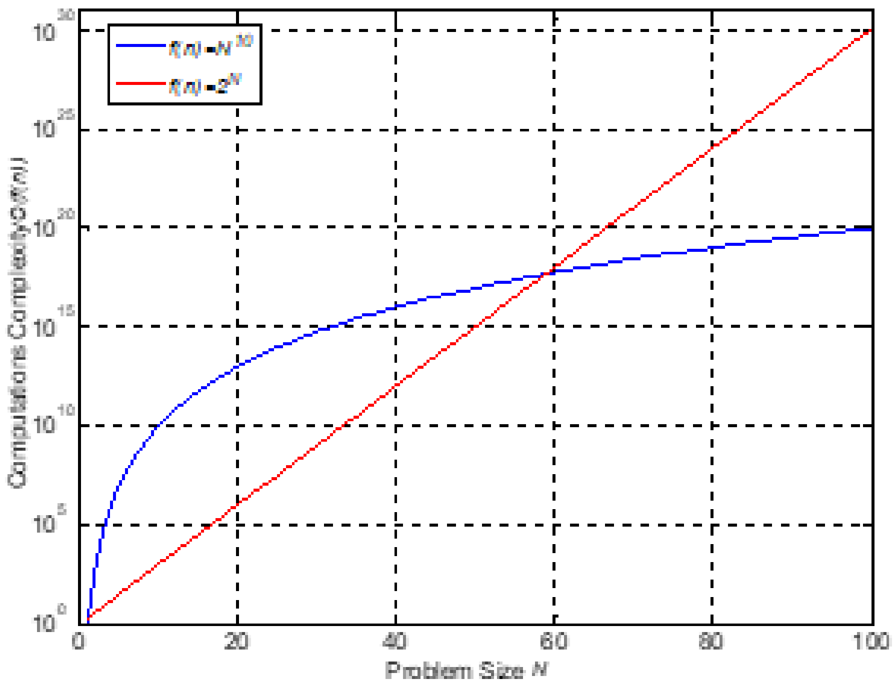
Approximation Algorithm

It is desirable for algorithms to run in **polynomial time**

Some problems are **not known to have** polynomial time solutions

Approximation algorithms run in **polynomial time** at the expense of rendering **sub-optimal** solutions

Polynomial vs Exponential Growth



Approximation Algorithm

- **Performance Ratio:**
- a ratio between the result obtained by the algorithm and the optimal cost or profit.
- Typically, this ratio is taken in whichever direction makes it bigger than one;
- for example, an algorithm that solves for a cost of \$2 an instance of a problem that has an optimal cost of \$1 has approximation ratio 2;
- but an algorithm that sells 10 airplane tickets (a profit of 10) when the optimum is 20 also has approximation ratio 2.

Vertex Cover Problem

- For the Graph G=(V, E). The subset S of V that meets every edge of E is called the vertex cover.

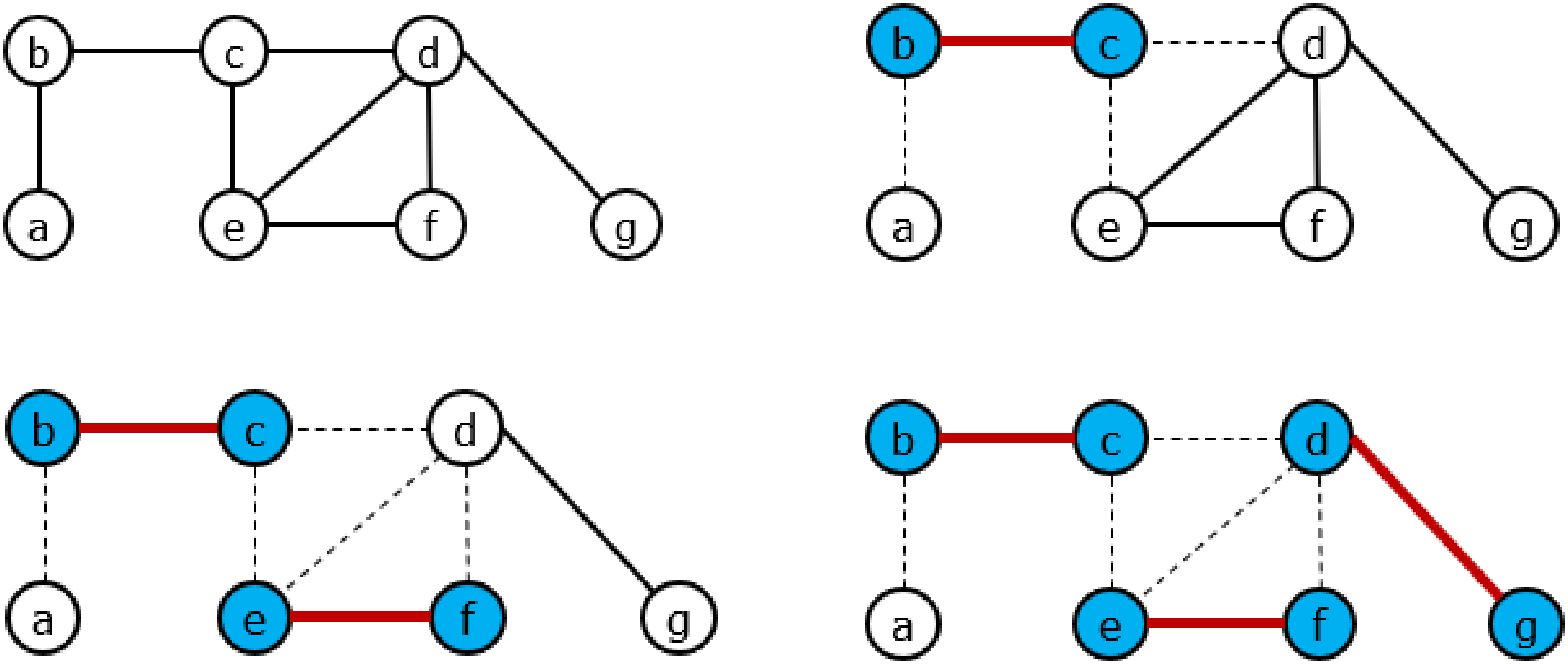
- The Vertex Cover problem is to find a vertex cover of the minimum size.

2-Approximate Vertex Cover Algorithm

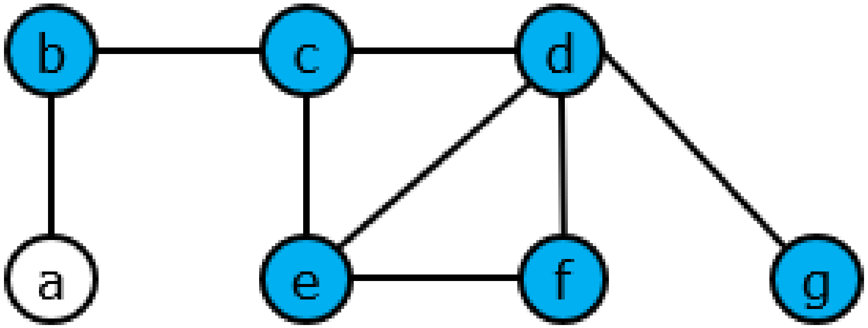
Algorithm 1 Vertex Cover 2-Approximation

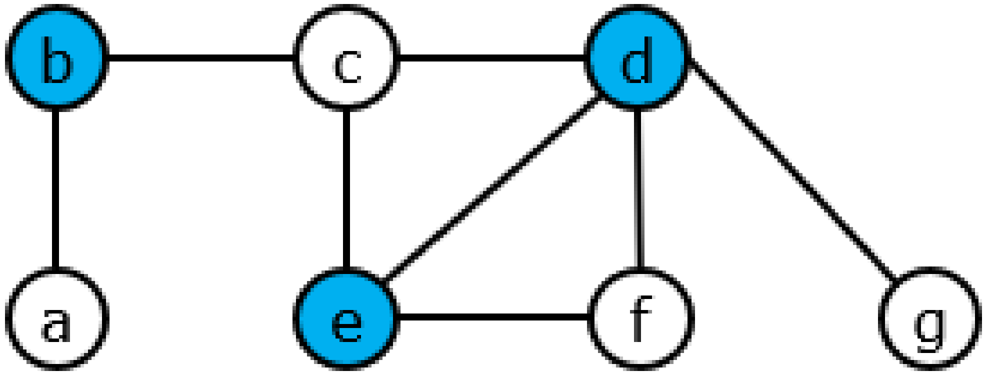
$U \leftarrow E$
$S \leftarrow \emptyset$
while U is not empty **do**
    Choose any $(v, w) \in U$.
    Add both v and w to S.
    Remove all edges adjoining v or w from U.
end while
return S

2-Approximate Vertex Cover Algorithm



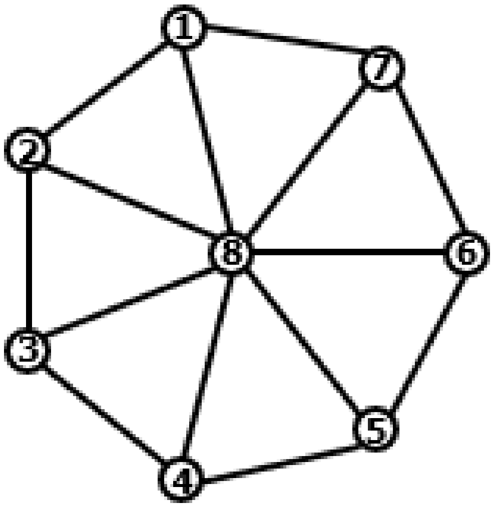
2-Approximate Vertex Cover Algorithm

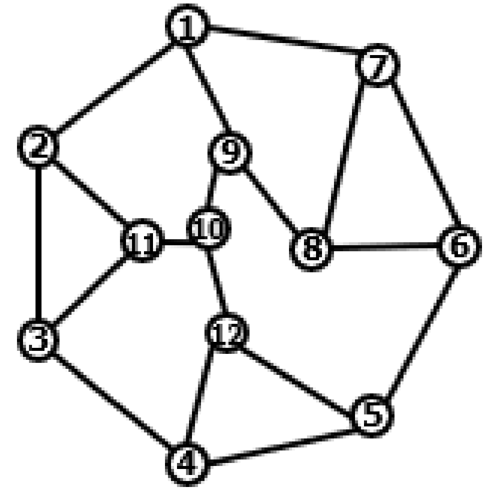




Performance Ratio?

Find Vertex Cover?



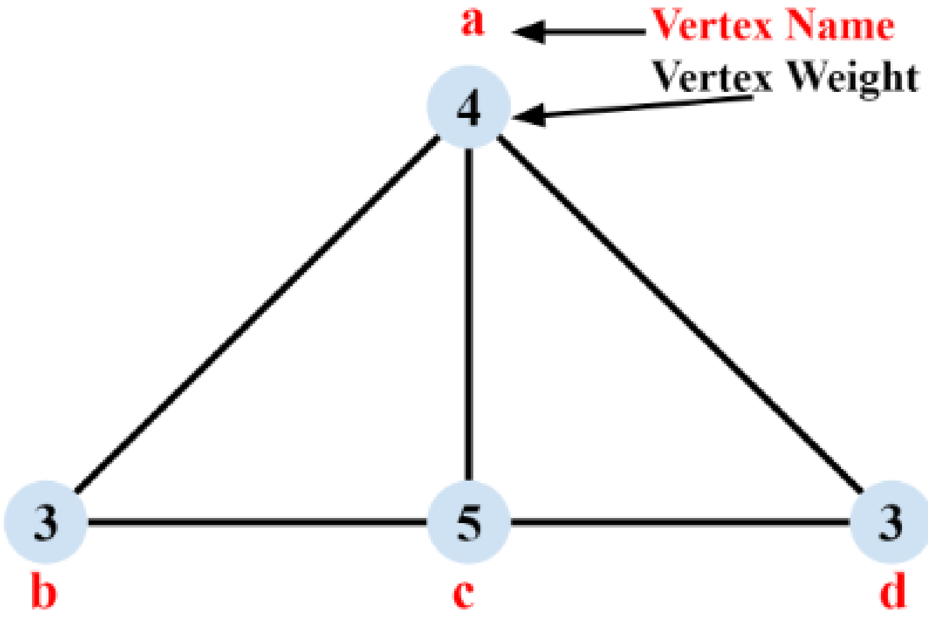


S={1, 3, 5, 6, 8}
|S| = 5

S={2, 4, 6, 7, 9, 11, 12}
|S| = 7

Weighted Vertex Cover

Given a graph $G = (V, E)$ *and a positive weight function* $w : V \rightarrow R^+$ *on the vertices, find a subset* $C \subseteq V$ *such that for all* $(u, v) \in E$*, at least one of* u *or* v *is contained in* C *and* $\sum_{v \in C} w(v)$ *is minimized.*



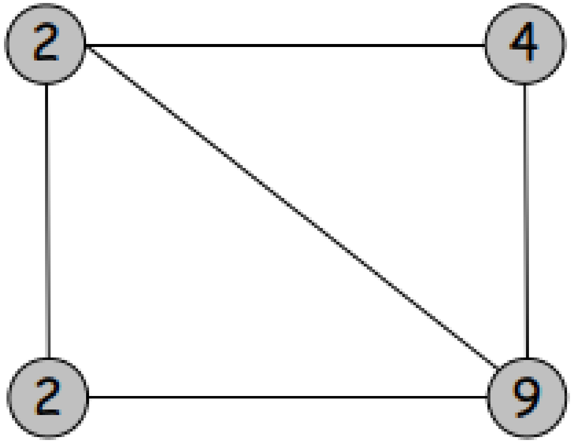
Weighted Vertex Cover: Pricing Method

Pricing method. Each edge must be covered by some vertex.

Edge $e = (i, j)$ pays price $p_e \geq 0$ to use vertex i and j.

Fairness. Edges incident to vertex i should pay $\leq w_i$ in total.

$$\text{for each vertex } i: \sum_{e=(i,j)} p_e \leq w_i$$



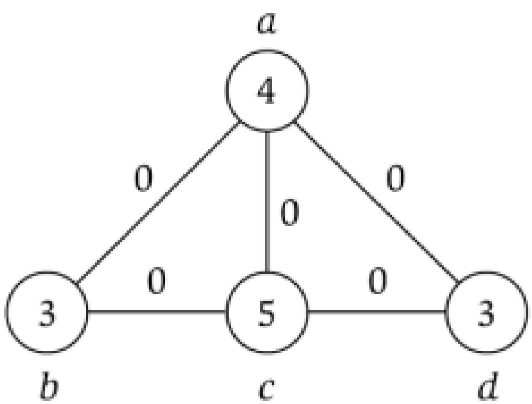
Weighted Vertex Cover: Pricing Method

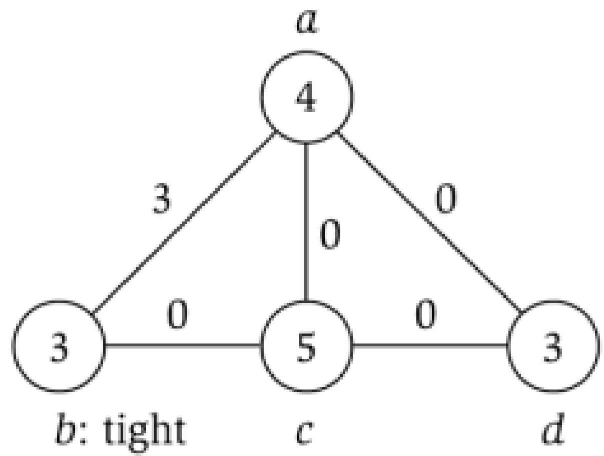
Pricing method. Set prices and find vertex cover simultaneously.

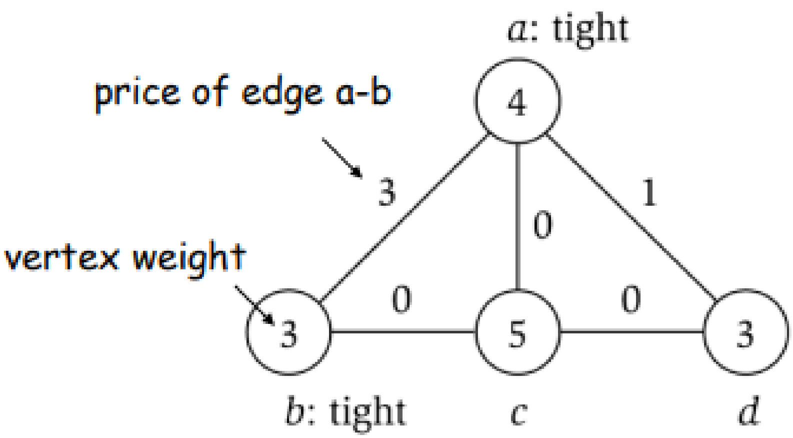
$$\sum_{e=(i,j)} p_e = w_i$$

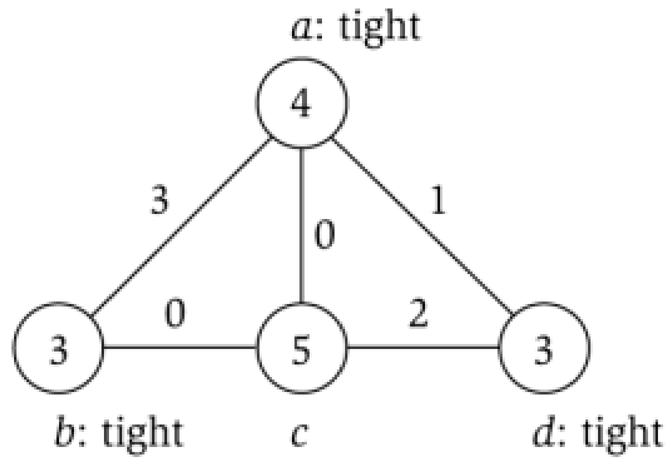
```
Weighted-Vertex-Cover-Approx(G, w) {
   foreach e in E
      pe = 0

   while (∃ edge i-j such that neither i nor j are tight)
      select such an edge e
      increase pe as much as possible until i or j tight
   }

   S ← set of all tight nodes
   return S
}
```

Weighted Vertex Cover: Pricing Method

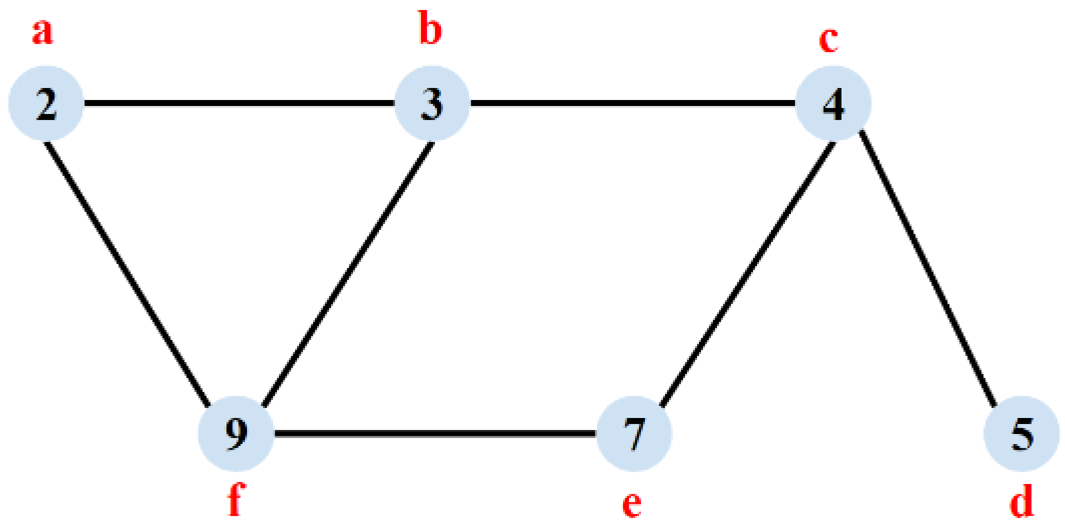








Weighted Vertex Cover: Pricing Method



Summary

- Matchings in Graphs