Report
Assignment 5

Name – Parul Bansal
Roll No – B14116

Question 1 (Vertex-Cover Problem)

Psuedo Code of k-node vertex cover

```
To search for a k -node vertex cover in G :
If G contains no edges, then the empty set is a vertex cover
If G contains > k |V| edges, then it has no k -node vertex cover
Else let e = (u, v) be an edge of G
    Recursively check if either of G −{u} or G −{v}
        has a vertex cover of size k − 1
    If neither of them does, then G has no k -node vertex cover
    Else, one of them (say, G −{u} ) has a (k − 1) - node vertex
    cover T
        In this case, T ∪ {u} is a k -node vertex cover of G
    Endif
Endif
```

Observations

- The running time of the Vertex Cover Algorithm on an n-node graph, with parameter k, is $O(2^k \cdot kn)$.

- The graph G has a vertex cover of size at most k if and only if at least one of the graphs $G - \{u\}$ and $G - \{v\}$ has a vertex cover of size at most $k - 1$.

Question 2 (Maximum-size independent set Problem)

Psuedo Code of max-size independent set in forest F

```
Let S be the independent set to be constructed (initially empty)
  While F has at least one edge
        Let e = (u, v) be an edge of F such that v is a leaf
        Add v to S
        Delete from F nodes u and v , and all edges incident to them
  Endwhile
Return S
```

Observations

- Time complexity of above algorithm is $O(n^2)$.
- If T = (V , E) is a tree and v is a leaf of the tree, then there exists a maximum-size independent set that contains v.
- It is a greedy algorithm

Question 3 (Maximum-weight independent set Problem)

Psuedo Code of max-weight independent set of a tree T

Root the tree at a node r
For all nodes u of T in post-order
  If u is a leaf then set the values:
    $M_{out}[u] = 0$
    $M_{in}[u] = w_u$
  Else set the values:
    $M_{out}[u] = \Sigma_{v\in children(u)}\ \max(M_{out}[u], M_{in}[u])$
    $M_{in}[u] = w_u + \Sigma_{v\in children(u)} M_{out}[u]$.
  Endif
Endfor
Return $\max(M_{out}[r], M_{in}[r])$

Observations

- The above algorithm finds a maximum-weight independent set in trees in linear time .
- Dynamic Programming is used in above algorithm.
- It is easy to recover an actual independent set of maximum weight by recording the decision we make for each node, and then tracing back through these decisions to determine which nodes should be included.