Chapter 53: polynomial-time bounded algorithm for Minimum Vertex Cover

Variable	Meaning
G	Input connected un-directed graph
X	Set of vertices
C	Final set of vertices

This is a polynomial algorithm for getting the minimum vertex cover of connected undirected graph. The time complexity of this algorithm is O(n2)

Section 53.1: Algorithm Pseudo Code

Algorithm PMinVertexCover (graph G)
Input connected graph G
Output Minimum Vertex Cover Set C

```
Set C <- new Set<Vertex>()

Set X <- new Set<Vertex>()

X <- G.getAllVerticiesArrangedDescendinglyByDegree()

for v in X do
    List<Vertex> adjacentVertices1 <- G.getAdjacent(v)

    if !C contains any of adjacentVertices1 then

        C.add(v)

for vertex in C do

    List<vertex> adjacentVertices2 <- G.adjacentVertecies(vertex)

    if C contains any of adjacentVertices2 then

        C.remove(vertex)


return C
```

> C is the minimum vertex cover of graph G

> we can use bucket sort for sorting the vertices according to its degree because the maximum value of degrees is (n-1) where n is the number of vertices then the time complexity of the sorting will be O(n)