NAME:	Shubham Solanki
UID:	2022301015
SUBJECT	Design and Analysis of Algorithms
EXPERIMENT NO:	9
AIM:	To implement Vertex Cover Problem

Algorithm:	**Vertex Cover Algorithm**
	
	Approx-Vertex-Cover (G = (V, E))
	
	1. {
	2. C = empty-set;
	3. E'= E;
	4. While E' is not empty do
	5. {
	6. Let (u, v) be any edge in E': (*)
	7. Add u and v to C;
	8. Remove from E' all edges incident to
	9. u or v;
	10. }
	11. Return C;
	12.}

Code	**Vertex Cover Problem** **Source Code** #include<iostream> #include <list> using namespace std;

```
class Graph
{
        int V;
        list<int> *adj;
public:
        Graph(int V);
        void addEdge(int v, int w);
        void printVertexCover();
};

Graph::Graph(int V)
{
        this->V = V;
        adj = new list<int>[V];
}

void Graph::addEdge(int v, int w)
{
        adj[v].push_back(w);
        adj[w].push_back(v);
}


void Graph::printVertexCover()
{

        bool visited[V];
        for (int i=0; i<V; i++)
        visited[i] = false;

        list<int>::iterator i;

        for (int u=0; u<V; u++)
        {

        if (visited[u] == false)
        {

        for (i= adj[u].begin(); i != adj[u].end(); ++i)
```

```
        {
                int v = *i;
                if (visited[v] == false)
                {

                visited[v] = true;
                visited[u]  = true;

                cout<<"\nConsidering age "<<u<<" and "<<v<<endl;
                break;
                }
        }
        }
        }

cout<<"\nThe Vertex Cover is as follows : ";
        for (int i=0; i<V; i++)

        {

        if (visited[i]) {

        cout << i << " ";
        }

        }
}


int main()
{
        int v;
        cout<<"\nEnter the number of vertices : ";
        cin>>v;

        Graph g(v);

        int e;
        cout<<"\nEnter the number of edges : ";
```

```
        cin>>e;
        cout<<endl;

        for(int i=0;i<e;i++){
        int a,b;
        cout<<"Enter the vertices for the  edge "<<i+1<<" :";
        cin>>a>>b;
        g.addEdge(a,b);
        }


        g.printVertexCover();

        cout<<endl<<endl;

        return 0;
}
```

Output

```
students@students-HP-280-G3-SFF-Business-PC:~/Desktop/daa/exp$ g++ verCover.cpp
students@students-HP-280-G3-SFF-Business-PC:~/Desktop/daa/exp$ ./a.out

Enter the number of vertices : 7

Enter the number of edges : 8

Enter the vertices for the  edge 1 :0 1
Enter the vertices for the  edge 2 :1 2
Enter the vertices for the  edge 3 :2 3
Enter the vertices for the  edge 4 :2 4
Enter the vertices for the  edge 5 :3 4
Enter the vertices for the  edge 6 :3 5
Enter the vertices for the  edge 7 :3 6
Enter the vertices for the  edge 8 :4 5

Considering age 0 and 1

Considering age 2 and 3

Considering age 4 and 5

The Vertex Cover is as follows : 0 1 2 3 4 5

students@students-HP-280-G3-SFF-Business-PC:~/Desktop/daa/exp$
```

Conclusion: Thus we have implemented vertex cover problem