P18-0126 AI LAB TASK

UCS using BFS:

```
def ucs(self, graph, start):
        visited, queue = set(), [start]
        while queue:
            vertex = queue.pop()  #vertex here represnts
            if vertex!='G':  #goal state found!!
                if vertex not in visited:
                    visited.add(vertex)
                    # new nodes are added to end of queue
                    queue.extend(graph[vertex] - visited)
            if vertex=='G':
                return path_cost
            return visited #if goad state visited then return the list! and the path cost will be counted
on the basis of list len() function!
```

IDFS using DFS:

```
def DFS(self,start,goal,maxDepth):

        if start == goal :
            return True

        if maxDepth <= 0 :
            return False

        for i in self.graph[src]:
            if(self.DFS(i,goal,maxDepth-1)):
                return True
        return False

    def IDDFS(self,start, goal, maxDepth):

        for i in range(maxDepth):
            if (self.DLS(start, goal, i)):

                return True
        return False
```