Program 3: Implement Iterative deepening search algorithm.

	PRANAY JAGADEESH IBM 18CS071 PROGRAM 3
+	PRANALY VANADELSA TOTAL
-	from collections import default dut
-	1/2.1 (24.11)
4	def - init - (self, verticer):
-1	21111 11 = 400 1514 00
-	seif graph = defaultdut (list)
-	sel ad Floo Bell V
	coll stalk w abberra
1	1. DIC/ goll is target, many spire
	if un == target return fille
	11 mooregal V
	for i in self graph [sec]:
	if (ref. D[S (i) target, max () Epih - 1)).
	neturn True
	neturn False
	del TDDFS (sell, src, target, max Depth).
	of felf. PLS (He , target, ").
	rebuen Tene
	peturn False
	g = Toraph (7);
	g = add Edge (0, 1)
4	g = add Edge (0,2)
4	g = add Edge (\$,3)
	g = add = Ige (1, 4)
	g = add Edge (2,5)
1	g = add Edge (2,6)
	target = 6; man Depth = 3; src = 0
1	if got DDFS (see, target, max Depth) == Toue:
	print ("Target is reachable from source "+
	cles: " within more depth")
	one print ("Target is NOT marhable for 11 1
	print ("Target is NOT reachable from source "+

```
Program:
from collections import defaultdict
class Graph:
       def __init__(self,vertices):
               self.V = vertices
               self.graph = defaultdict(list)
       def addEdge(self,u,v):
               self.graph[u].append(v)
       def DLS(self,src,target,maxDepth):
               if src == target : return True
               if maxDepth \le 0: return False
               for i in self.graph[src]:
                              if (self.DLS (i, target, maxDepth-1)):\\
                                      return True
               return False
       def IDDFS(self,src, target, maxDepth):
```

```
for i in range(maxDepth):
                    if (self.DLS(src, target, i)):
                          return True
             return False
g = Graph(7);
g.addEdge(0, 1)
g.addEdge(0, 2)
g.addEdge(1, 3)
g.addEdge(1, 4)
g.addEdge(2, 5)
g.addEdge(2, 6)
target = 6; maxDepth = 3; src = 0
if g.IDDFS(src, target, maxDepth) == True:
      print ("Target is reachable from source " +
             "within max depth")
else:
      print ("Target is NOT reachable from source " +
             "within max depth")
Output:
 Target is reachable from source within max depth
 Process finished with exit code 0
```