listofNodes ← [ ]

CLASS newVertex

INIT(value)

value ← value

APPEND (value) TO listofNodes

CLASS Graph

INIT(nodes)

nodes ← {}

FOR i IN Nodes

self.nodes[i] ← []

edges ← DICT

DEF add\_node(value)

nodes[value] ← []

DEF add\_edge(from\_node, to\_node)

APPEND (to\_node) TO edges[from\_node]

APPEND (from\_node) TO edges[to\_node]

myList ← ['A', 'B', 'C', 'D', 'E', 'F', 'G']

FOR element IN myList:

newVertex(element)

myGraph ← Graph(listofNodes)

myGraph.add\_node('H')

myGraph.add\_edge('A', 'B')

myGraph.add\_edge('A', 'C')

myGraph.add\_edge('B', 'D')

OUTPUT(myGraph.nodes)