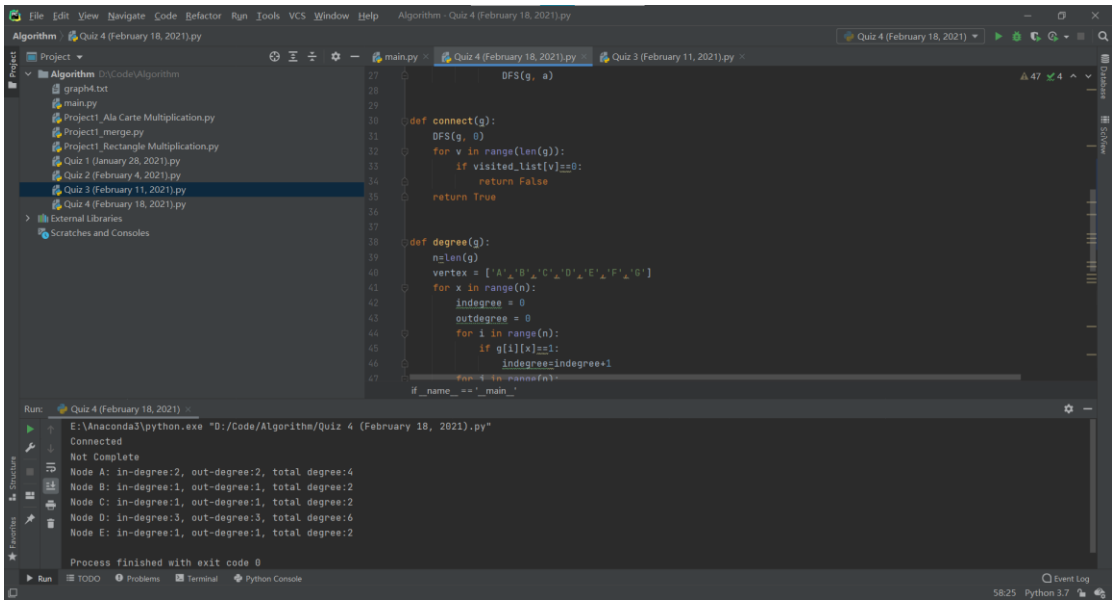
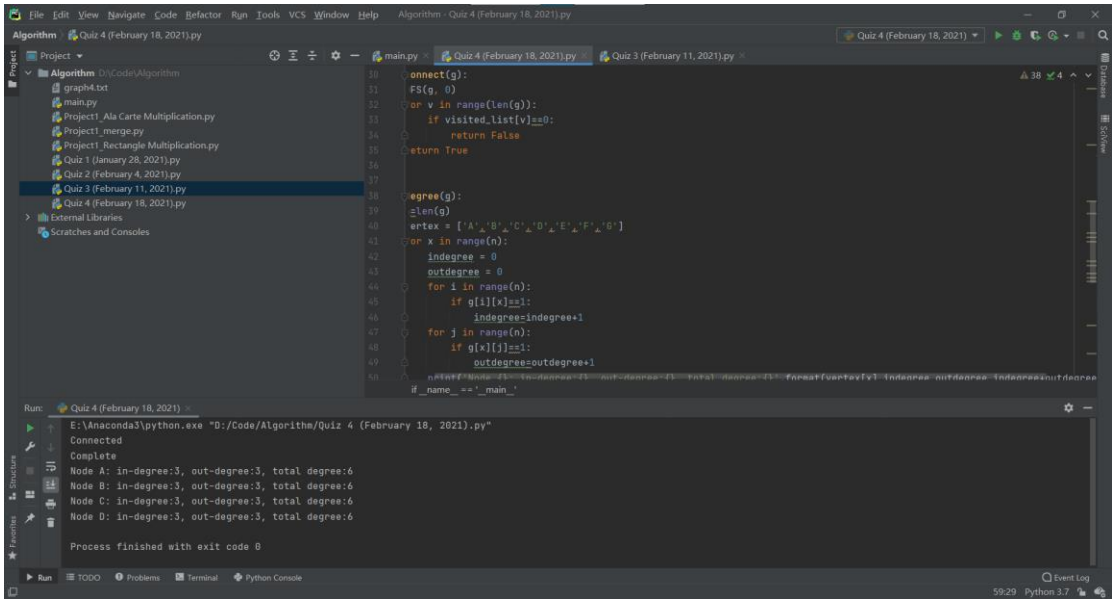
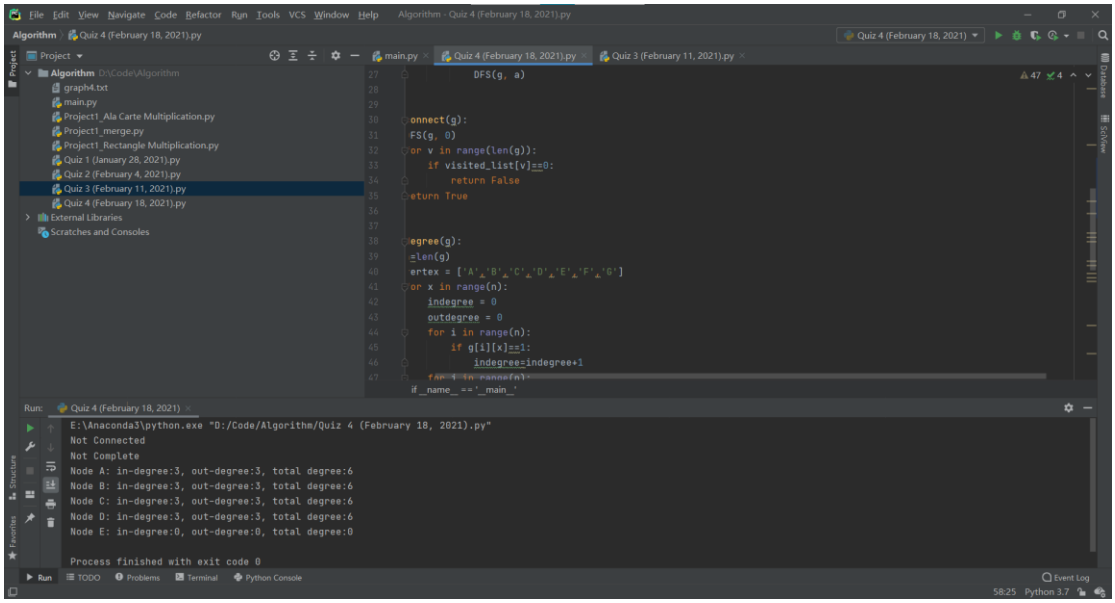
Test case1:



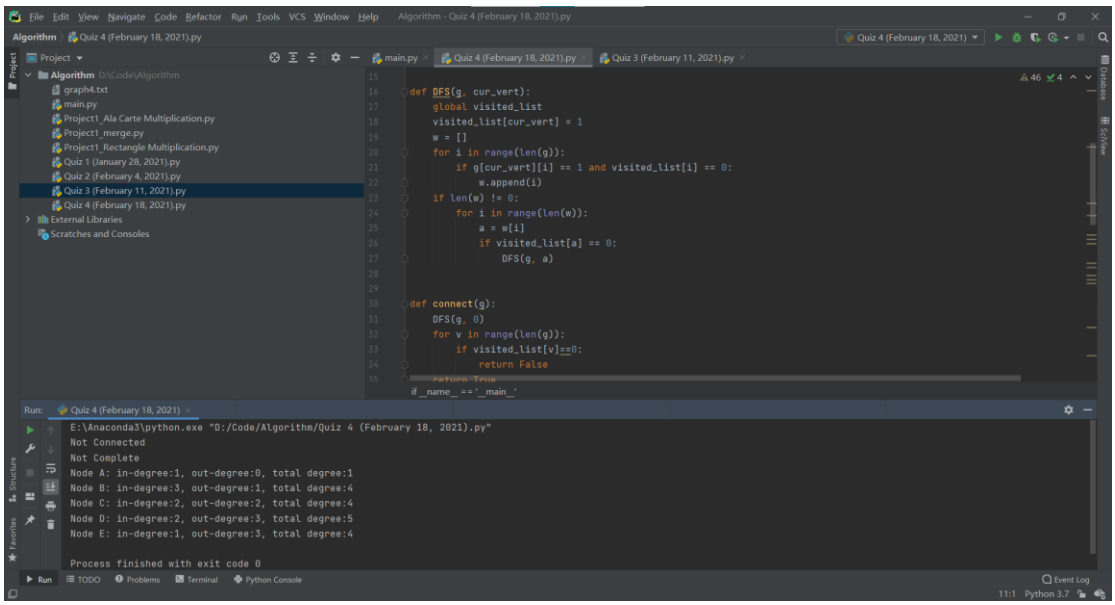
Test case2:



Test case3:


DFS(g, a)
onnect(g):
FS(g, 0)
or v in range(len(g)):
if visited_list[v]==0:
return False
eturn True
egree(g):
=len(g)
ertex = ['A','B','C','D','E','F','G']
or x in range(n):
indegree = 0
outdegree = 0
for i in range(n):
if g[i][x]==1:
indegree=indegree+1
if __name__ == '__main__'
E:\Anaconda3\python.exe "D:/Code/Algorithm/Quiz 4 (February 18, 2021).py"
Not Connected
Not Complete
Node A: in-degree:3, out-degree:3, total degree:6
Node B: in-degree:3, out-degree:3, total degree:6
Node C: in-degree:3, out-degree:3, total degree:6
Node D: in-degree:3, out-degree:3, total degree:6
Node E: in-degree:0, out-degree:0, total degree:0
Process finished with exit code 0

Test case4:


def DFS(g, cur_vert):
global visited_list
visited_list[cur_vert] = 1
w = []
for i in range(len(g)):
if g[cur_vert][i] == 1 and visited_list[i] == 0:
w.append(i)
if len(w) != 0:
for i in range(len(w)):
a = w[i]
if visited_list[a] == 0:
DFS(g, a)
def connect(g):
DFS(g, 0)
for v in range(len(g)):
if visited_list[v]==0:
return False
if __name__ == '__main__'
E:\Anaconda3\python.exe "D:/Code/Algorithm/Quiz 4 (February 18, 2021).py"
Not Connected
Not Complete
Node A: in-degree:1, out-degree:0, total degree:1
Node B: in-degree:3, out-degree:1, total degree:4
Node C: in-degree:2, out-degree:2, total degree:4
Node D: in-degree:2, out-degree:3, total degree:5
Node E: in-degree:1, out-degree:3, total degree:4
Process finished with exit code 0