



Prepare

Certify

Compete

Apply



Search



adrian\_santacruz ▾

[All Contests](#) > [HW5\\_Graphs](#) > [H5\\_1\)Connectivity](#)

# H5\_1)Connectivity

Problem

Submissions

Leaderboard

Submitted 6 hours ago • Score: 35.00

Status: **Accepted**

Success 0.07s

✓	Test Case #0	✓	Test Case #1	✓	Test Case #2
✓	Test Case #3	✓	Test Case #4	✓	Test Case #5
✓	Test Case #6	✓	Test Case #7	✓	Test Case #8
✓	Test Case #9				

## Submitted Code

Language: Python 3

[Open in editor](#)

```
1 def getNum(letter):
2     return ord(letter) - ord("A")
3
4 def getLetter(num):
5     return chr(num + ord("A"))
6
7 def AdjacencyMatrix(V, edges):
8     graph = [[False]*V for _ in range(V) ]
9
10    for edge in edges:
11        X,Y = (edge)
12        i, j = getNum(X),getNum(Y)
13
14        graph[i][j] = True
15        graph[j][i] = True
16    return graph
17
18 def firstFalse(L):
19     i = 0
20     while L[i] == True:
21         i = i + 1
22     return i
23
24 def connectivity(V, edges):
25
26     M = AdjacencyMatrix(V, edges)
27     visitedbool = [False]*V
28     subgraphs = []
29
30     while not all(visitedbool):
31         subgraph = []
32         start = firstFalse(visitedbool)
```

```
33     stack = [start]
34
35     while stack:
36         node = stack.pop()
37         if not visitedbool[node]:
38             visitedbool[node] = True
39             subgraph.append(node)
40             for i in range(V):
41                 if M[node][i] == True and not visitedbool[i]:
42                     stack.append(i)
43
44     subgraph.sort()
45     subgraph = [getLetter(i) for i in subgraph]
46     word = ""
47     for letter in subgraph:
48         word = word + letter
49
50     subgraphs.append(word)
51
52     if len(subgraphs) == 1:
53         print("T")
54     else:
55         print("F")
56     for g in subgraphs:
57         print(g)
58
59 numbers = input()
60 numbers = tuple(map(int,numbers.split()))
61 V, E = numbers
62
63 edges = []
64 for i in range(E):
65     edge = input()
66     edge = tuple(map(str,edge.split()))
67     edges.append(edge)
68
69 connectivity(V, edges)
70
71
```