

Actividad de clase 16 - Caminos con BFS

Victor Misael Escalante Alvarado, A01741176

Foto del Código funcionando:

The screenshot displays a Jupyter Notebook environment with a Python script named `Actividadenclass16.py`. The script implements a graph search algorithm, likely Breadth-First Search (BFS), to find the shortest path between two nodes in a graph.

Script Content:

```

1 from collections import deque, defaultdict
2
3 def bfs(graph, start, end):
4     # Inicializamos la lista con el nodo de inicio, la lista de nodos visitados y la lista de pesos
5     queue = deque([(start, [start], [0])])
6     visited = set()
7
8     # Muestra la cola no este vacia
9     while queue:
10        # Sacamos el nodo actual, el camino y los pesos
11        current_node, path, weights = queue.popleft()
12        # Si current_node no es visitado:
13        if current_node not in visited:
14            continue
15        visited.add(current_node)
16
17        # Si current_node == end:
18        if current_node == end:
19            return path, weights[0]
20
21        for neighbor, weight in graph[current_node]:
22            if neighbor not in visited:
23                queue.append((neighbor, path + [neighbor], weights + [weight]))
24
25    return [], []
26
27 # Gráfico prueba:
28 graph_prueba = {
29     'A': ['B', 3], ['C', 4]},
30     'B': ['C', 2], ['D', 5]},
31     'C': ['D', 1]},
32     'D': []
33 }
34
35 # Lectura de un archivo
36 graph = defaultdict(list)
37 with open('txt/graph.txt') as file:
38     # Ignorar la primera línea
39     # En la primera línea se encuentra el número de nodos
40     nodos = file.readline()
41     for i in range(int(nodos)):
42         graph[str(i+1)] = []
43
44 # Formato de la línea nodo nodo peso
45 for line in file:
46     start, end, weight = line.split()
47     weight = int(weight)
48     graph[start].append((end, weight))
49
50 start = 'A'
51 end = 'D'

```

Execution Output:

```

Python 3.12.5 64-bit
FileNotFoundError: [Errno 2] No such file or directory: 'txt/graph.txt'
Traceback (most recent call last):
  File "/Users/misa_v/Library/CloudStorage/OneDrive-InstitutoTecnológicodeEstudiosSuperioresdeMonterrey/Sem 5/Algoritmos_Scripts/Python/Actividadenclass16.py", line 37, in <module>
    with open('txt/graph.txt') as file:
         ^^^^^^^^^^^^^^^^^^^^^
FileNotFoundError: [Errno 2] No such file or directory: 'txt/graph.txt'
TypeError: 'str' object cannot be interpreted as an integer
Traceback (most recent call last):
  File "/Users/misa_v/Library/CloudStorage/OneDrive-InstitutoTecnológicodeEstudiosSuperioresdeMonterrey/Sem 5/Algoritmos_Scripts/Python/Actividadenclass16.py", line 46, in <module>
    for line in file:
         ^^^^^^^^^^^
FileNotFoundError: [Errno 2] No such file or directory: 'txt/graph.txt'

```

The output shows several errors: a `FileNotFoundError` for the missing `txt/graph.txt` file, and a `TypeError` indicating that a string object cannot be interpreted as an integer, likely due to the `int(nodos)` conversion in the loop.

Enlace a COLAB:

[\[Enlace a Google COLAB\]](#)