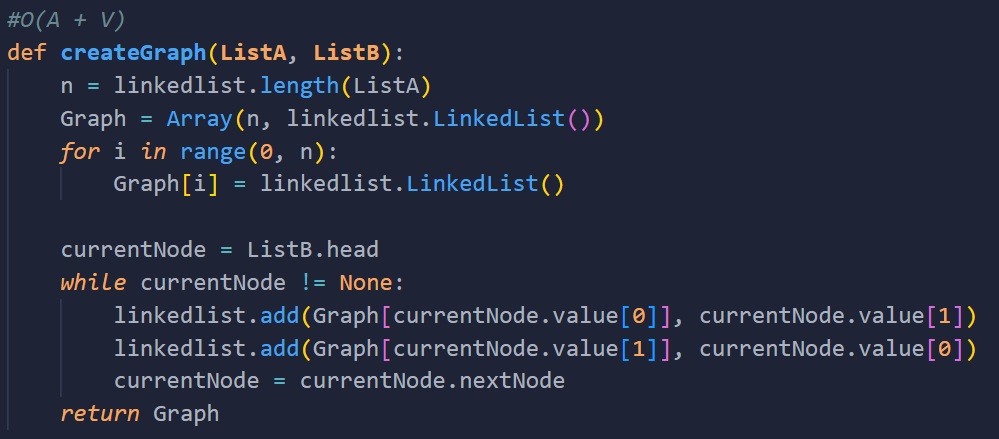
**Trabajo Práctico**

**Grafos**

**Tomás Rando - 14004 - LCC**

**1)**

**createGraph(List1, List2)**



**2)**

**existPath(Grafo, v1, v2)**



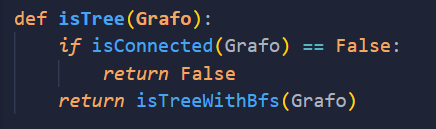
**3)**

**isConnected(Grafo)**



**4)**

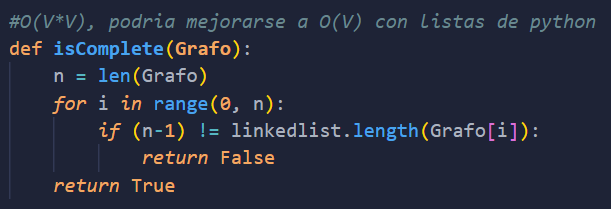
**isTree(Grafo)**





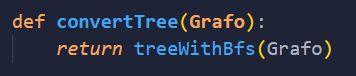
**5)**

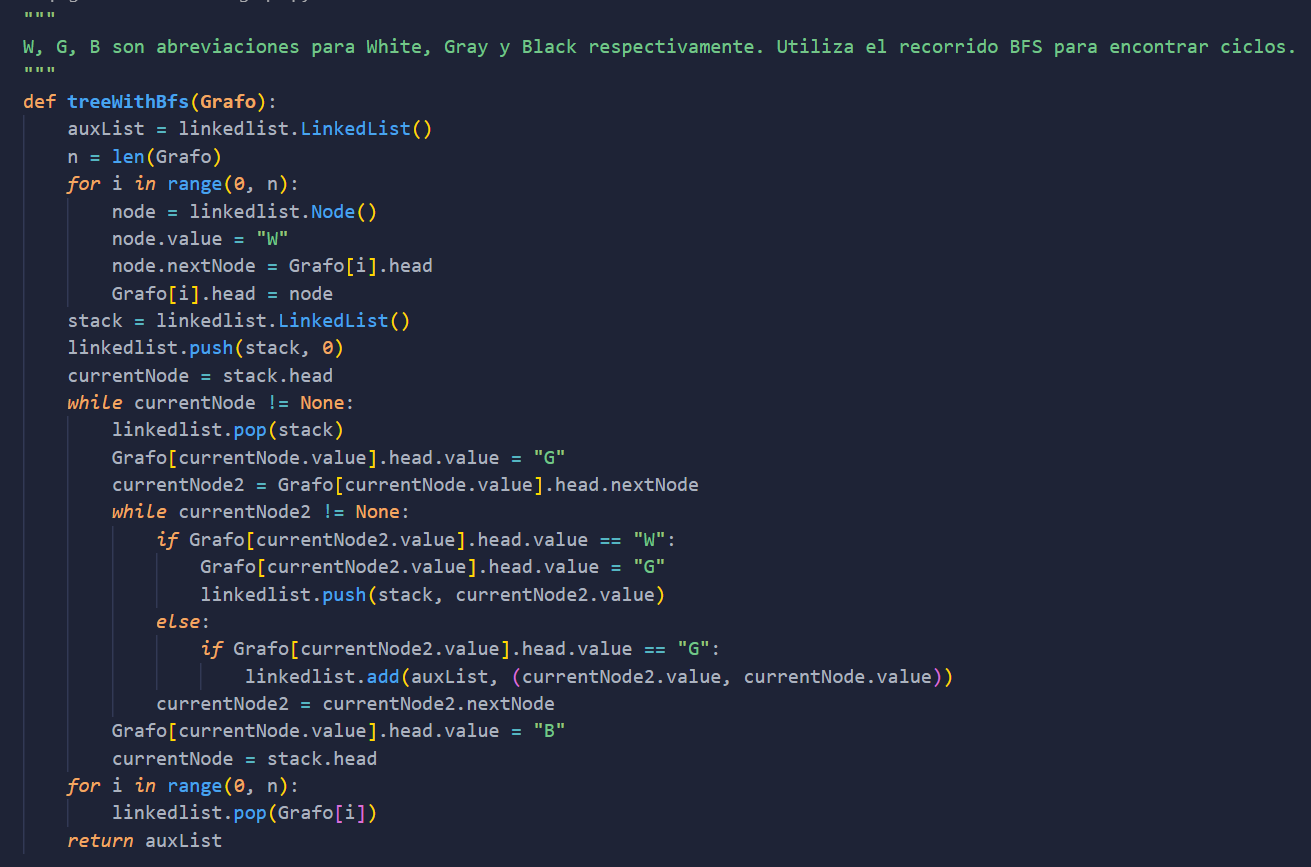
**isComplete(Grafo)**



**6)**

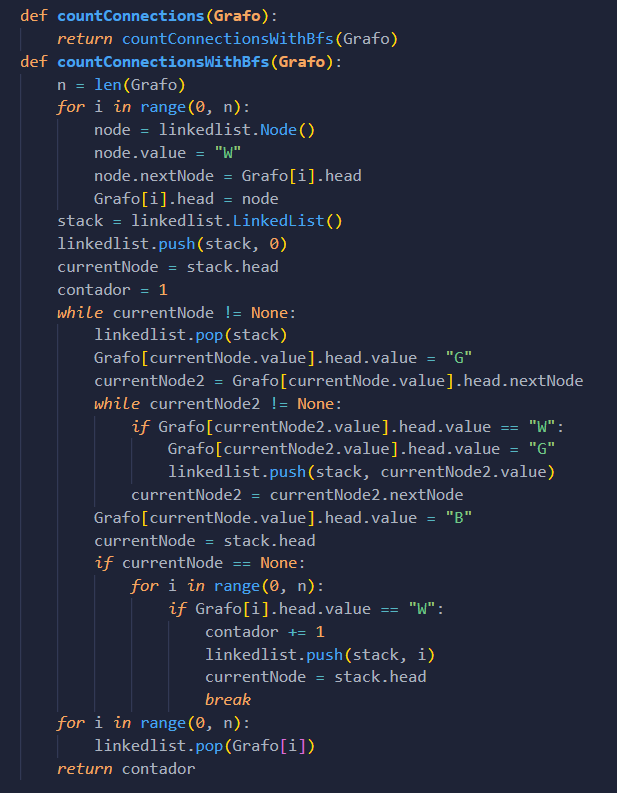
**convertTree(Grafo)**





**7)**

**countConnections(Grafo)**



**convertToBFSTree(Grafo, v)**

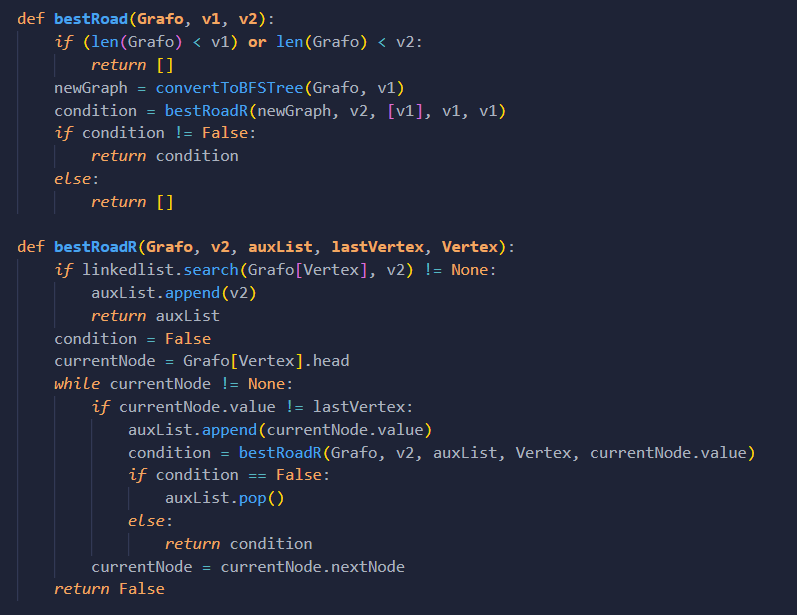


**convertToDFSTree(Grafo, v)**



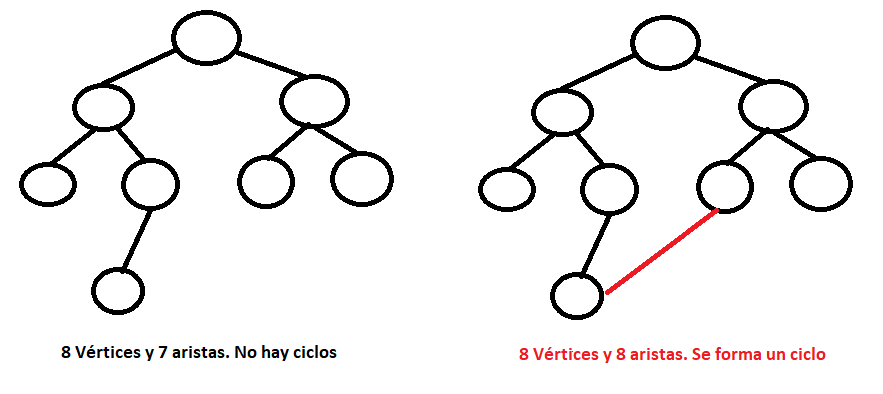
**10)**

**bestRoad(Grafo, v1, v2)**



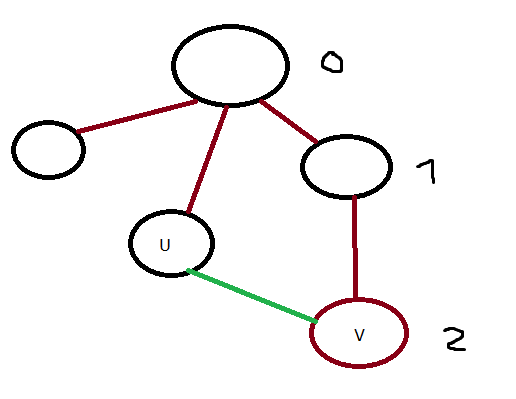
**12)**

**Por propiedad, un grafo de n vértices tiene n-1 aristas en total. Si tenemos un árbol normal de m vértices, entonces este tendrá m - 1 aristas. Si le agregamos una arista más, obtenemos m aristas, por lo que no se cumple la propiedad y es falso.**



**13)**

**Si una arista está conectada tanto a v como a u, eso significa que v está un nivel anterior o superior al de u. El algoritmo BFS recorre el grafo por niveles, añadiendo las aristas al árbol teniendo en cuenta eso. Si una arista (v,u) no ha sido agregada, significa que el vértice v fue encontrado añadiendo los vértices adyacentes a otro vértice del nivel de u y que se encuentra en el nivel inferior a este. Puesto que si no lo estuviese, hubiese sido agregado al recorrer u, y la arista (v, u) si existiría en el árbol BFS.**



**Por ejemplo, en este árbol, la arista (u,v) (La verde), no fue agregada porque el BFS recorrió añadiendo los vértices por niveles, y en el nivel 1 recorrió desde los vértices de la derecha hacia la izquierda y se encontró una arista que une a v. Por lo explicado anteriormente, el nivel del vértice u y v, difiere en 1 a lo sumo, puesto que de lo contrario, hubiese sido agregada al agregar las aristas de u, ya que va recorriendo por niveles.**