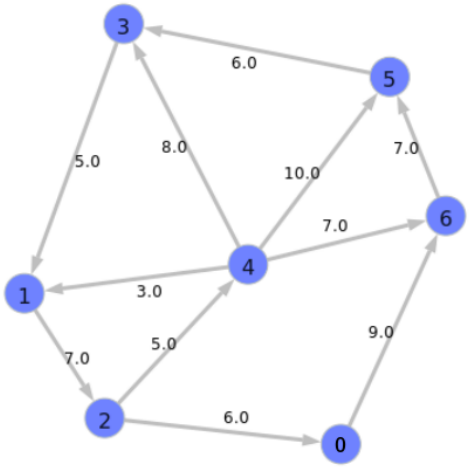
Documentação para o trabalho de Grafos

Para testar utilizei o Grafo G abaixo, sendo retirado do Slide e mudando o número 7 para 0.

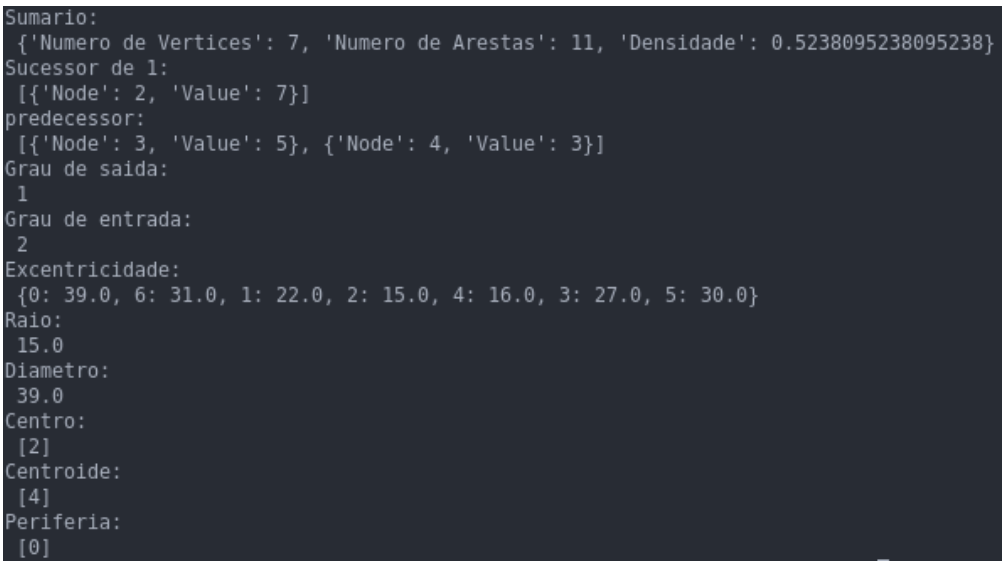


Este exemplo está configurado no arquivo nodes.txt para facilitar o carregamento e configuração de G.

O código escrito é este abaixo:

```
from graph import Graph
from graph_utils import *

#criando grafo
graph = Graph()

#carregando o grafo de um arquivo de texto
graph.load('nodes.txt')

#criando matriz de distancia
floyd_warshall(graph)

#mostrando informações sobre o grafo
print("Sumario:\n",summary(graph))
print("Sucessor de 1:\n",successor(graph,1))
print("predecessor:\n",predecessor(graph,1))
print("Grau de saida:\n",outdegree(graph,1))
print("Grau de entrada:\n",indegree(graph,1))
print("Excentricidade:\n",eccentricity(graph))
print("Raio:\n",radius(graph))
print("Diametro:\n",diameter(graph))
print("Centro:\n",center(graph))
print("Centroide:\n",centroid(graph))
print("Periferia:\n",periphery(graph))
```

E no output obtivemos:



Para utilizar as funções é necessário como argumento apenas o grafo criado e que seja utilizado a função que cria a matriz de distâncias anteriormente.

eccentricity retorna um dict com a excentricidade de cada nó.
radius retorna o raio do grafo.
diameter retorna o diâmetro do grafo.
center retorna os nós que estão no centro.
centroid retorna os centróides do grafo
periphery retorna os nós que estão na periferia do grafo.