

```

import networkx as nx
import matplotlib.pyplot as plt

def premier(atester):
    k = 2
    if atester in [0, 1]: return False
    if atester in [2, 3, 5, 7]: return True
    while True:
        if k * k > atester: return True
        else:
            if atester % k == 0: return False
            else: k = k + 1

for n in range(98,100,2):
    L=[0]
    for k in range(3,n-1,2):
        if premier(k):
            L.append(k)
    L.append(n)
    print('L = ',L)
    ecarts = []
    for indice in range(1,len(L)):
        ecarts.append(L[indice]-L[indice-1])
    print('ecarts = ',ecarts)
    poids_fixes = ecarts
    SIZE = len(poids_fixes) + 1
    G_full = nx.grid_2d_graph(SIZE, SIZE)
    nodes_to_keep = [(i, j) for (i, j) in G_full.nodes() if i >= j]
    G_complet = G_full.subgraph(nodes_to_keep).copy()
    for (u, v) in G_complet.edges():
        (r1, c1), (r2, c2) = u, v
        if r1 == r2: # Horizontale
            G_complet.edges[u, v]['weight'] = poids_fixes[min(c1, c2)]
        else: # Verticale
            G_complet.edges[u, v]['weight'] = poids_fixes[min(r1, r2)]
    node_isole = (SIZE - 1, 0) # Le coin bas-gauche
    G_gros = G_complet.copy()
    if node_isole in G_gros:
        G_gros.remove_node(node_isole) # Supprime le noeud ET ses aretes
    path_lengths = dict(nx.all_pairs_dijkstra_path_length(G_gros,
weight='weight'))
    eccs_gros = {node: max(dists.values()) for node, dists in
path_lengths.items()}
    min_ecc = min(eccs_gros.values())
    centers_triangle = [n for n, ecc in eccs_gros.items() if ecc == min_ecc]
    plt.figure(figsize=(10, 8))
    pos = {(i, j): (j, -i) for (i, j) in G_complet.nodes()}
    node_labels = {n: f"{eccs_gros[n]}" for n in G_gros.nodes()}
    node_labels[node_isole] = "X" # Marquer le noeud exclu
    node_colors = []
    for node in G_complet.nodes():
        if node == node_isole:
            node_colors.append('black') # Le point exclu est noir
        elif node in centers_triangle:
            node_colors.append('cyan') # Les centres du "gros" sont cyan
        else:
            node_colors.append('lightgrey') # Le reste est gris
    nx.draw_networkx_edges(G_complet, pos, alpha=0.5, edge_color='gray')
    edge_labels = nx.get_edge_attributes(G_complet, 'weight')
    nx.draw_networkx_edge_labels(G_complet, pos, edge_labels=edge_labels,
font_color='red', font_size=8)
    nx.draw_networkx_nodes(G_complet, pos,
node_color=node_colors,
node_size=800)

```

```
nx.draw_networkx_labels(G_complet, pos, labels=node_labels, font_size=8)
plt.title(f"Triangle {SIZE}x{SIZE} : Coin exclu (X, noir)\nCentres du graphe
triangulaire sans pointe en CYAN")
plt.axis('off')
#plt.show()
print(f"Le(s) centre(s) du graphe triangulaire sans pointe est/sont :
{centers_triangle}")
for sommet in centers_triangle:
    print(n, '=', L[sommet[1]], '+', n-L[sommet[1]], ' ', end='')
nomfic = 'triangle'+str(n)
plt.savefig(nomfic)
plt.close()
print('')
```