Documentazione relativa alla smulazione del Protocollo di Routering

Progetto Python: Simulazione di Protocollo di Routing

- Descrizione: Creare uno script Python che simuli un protocollo di routing semplice, come il Distance Vector Routing. Gli studenti implementeranno gli aggiornamenti di routing tra i nodi, con il calcolo delle rotte più brevi.
- Obiettivi: Implementare la logica di aggiornamento delle rotte, gestione delle tabelle di routing e calcolo delle distanze tra nodi.
- Consegne richieste: Codice Python ben documentato, output delle tabelle di routing per ogni nodo e relazione finale che spieghi il funzionamento dello script.

Classi

`Node`

Rappresenta un singolo nodo nella rete. Ogni nodo mantiene un vettore di distanza (DV) e un elenco dei suoi vicini.

Attributi:

- `name` (`str`): Nome del nodo.
- `distanceVector` (`dict`): La tabella di routing del nodo, dove le chiavi sono i nomi delle destinazioni e i valori sono le distanze più brevi conosciute.
- `neighbors` (`dict`): Dizionario dei vicini con la relativa distanza. Ogni voce contiene:
  - `'node'`: L'oggetto del nodo vicino.
  - `'distance'`: La distanza al vicino.

Metodi:

1. `add_neighbor(self, neighbor, distance)`
   Aggiunge un vicino al nodo. Inizializza il vicino nel `distanceVector`.
   - `neighbor` (`Node`): Oggetto del nodo vicino.
   - `distance` (`int`): Distanza al vicino.
2. `update_distanceVector(self)`
   Aggiorna il vettore di distanza del nodo basandosi sui vettori ricevuti dai vicini.
   - **Restituisce**: `True` se il `distanceVector` è stato aggiornato, altrimenti `False`.
3. `send_distance_vector(self)`
   Invia il vettore di distanza corrente del nodo ai vicini.
   - **Restituisce**: Il `distanceVector`.

4. `print_distanceVector(self)`
   Stampa il vettore di distanza corrente in un formato leggibile.

`RoutingNetwork`

Rappresenta l'intera rete di nodi e gestisce le connessioni e l'esecuzione del protocollo di routing.

Attributi:

- `nodes` (`dict`): Un dizionario dove le chiavi sono i nomi dei nodi e i valori sono oggetti `Node`.

Metodi:

1. `add_node(self, node)`
   Aggiunge un nodo alla rete.
   - `node` (`Node`): Il nodo da aggiungere.
2. `connect_nodes(self, node1, node2, distance)`
   Connette due nodi con una distanza specifica.
   - `node1` (`Node`): Primo nodo.
   - `node2` (`Node`): Secondo nodo.
   - `distance` (`int`): Distanza tra i nodi.
3. `routing_protocol(self)`
   Implementa il protocollo di routing basato sui vettori di distanza. I nodi scambiano i propri vettori iterativamente fino alla convergenza.
   - **Processo di esecuzione**:
     1. Ogni nodo invia il proprio vettore di distanza ai vicini.
     2. Ogni nodo aggiorna il proprio vettore di distanza usando l'equazione di Bellman-Ford: [D_{n,d} = \min(D_{n,d}, D_{n,v} + D_{v,d})] dove (D_{n,d}) è la distanza dal nodo (n) alla destinazione (d), e (v) è un vicino.
     3. I print mostrano lo stato delle tabelle di routing prima e dopo ogni aggiornamento.
     4. Il processo termina quando non ci sono ulteriori aggiornamenti (convergenza).

Utilizzo

Configurazione della Rete

Per configurare opportunamente la rete è necessario modificare il main per ottenere la topografia desiderata, nel main fornito troviamo tre esempi di topografie ordinate per complessità cresciente, decommentando opportunamente il codice di una di esse e commentando quelle non desiderate si può osservare l'algorimo all'opera. Se si desidera configurare una propria topografia di rete si può prendere come esempio la topografia n. 2 qui riportata:

```
# Creazione di una rete
network = RoutingNetwork()
```

```
# Creazione dei nodi
A = Node('A')
B = Node('B')
C = Node('C')
D = Node('D')
E = Node('E')
F = Node('F')

# Aggiunta dei nodi alla rete
network.add_node(A)
network.add_node(B)
network.add_node(C)
network.add_node(D)
network.add_node(E)
network.add_node(F)

# Connessione tra nodi con distanze
network.connect_nodes(A, B, 2)
network.connect_nodes(A, C, 2)
network.connect_nodes(A, D, 9)
network.connect_nodes(A, F, 5)
network.connect_nodes(B, D, 3)
network.connect_nodes(C, D, 5)
network.connect_nodes(C, F, 11)
network.connect_nodes(D, E, 3)
network.connect_nodes(E, C, 1)
network.connect_nodes(E, F, 7)
```

Una volta configurata opportunamente la topografia desiderata lanciando lo script esso restituirà a terminale la tabella definitiva arrivandoci passo a passo.

Esempio di Output

Durante l'esecuzione del protocollo, il programma fornisce print dettagliati, tra cui:

- La tabella di routing di ogni nodo prima e dopo ogni aggiornamento in ciascuna iterazione.
- Notifiche quando un nodo riceve un vettore di distanza da un vicino.
- Tabelle di routing finali dopo il raggiungimento della convergenza.

Iterazione Campione:

Possiamo di seguito osservare l'output ottenuto a terminale dalla topografia di rete n. 2

```
-------------------------- Iterazione: 1 --------------------------
------------------------- A riceve DV(B) -------------------------
----------------------------- Tabella -----------------------------
A: {'A': 0, 'B': 2, 'C': 2, 'D': 9, 'F': 5}
------------------------ Tabella aggiornata ------------------------
A: {'A': 0, 'B': 2, 'C': 2, 'D': 5, 'F': 5, 'E': 3}
--------------------------------------------------------------------
------------------------- A riceve DV(C) -------------------------
```

```
---------------------------- Tabella ----------------------------
A: {'A': 0, 'B': 2, 'C': 2, 'D': 5, 'F': 5, 'E': 3}
------------------------ Tabella aggiornata ------------------------
A: {'A': 0, 'B': 2, 'C': 2, 'D': 5, 'F': 5, 'E': 3}
--------------------------------------------------------------------
------------------------ A riceve DV(D) ------------------------
---------------------------- Tabella ----------------------------
A: {'A': 0, 'B': 2, 'C': 2, 'D': 5, 'F': 5, 'E': 3}
------------------------ Tabella aggiornata ------------------------
A: {'A': 0, 'B': 2, 'C': 2, 'D': 5, 'F': 5, 'E': 3}
--------------------------------------------------------------------
------------------------ A riceve DV(F) ------------------------
---------------------------- Tabella ----------------------------
A: {'A': 0, 'B': 2, 'C': 2, 'D': 5, 'F': 5, 'E': 3}
------------------------ Tabella aggiornata ------------------------
A: {'A': 0, 'B': 2, 'C': 2, 'D': 5, 'F': 5, 'E': 3}
--------------------------------------------------------------------
------------------------ B riceve DV(A) ------------------------
---------------------------- Tabella ----------------------------
B: {'B': 0, 'A': 2, 'D': 3}
------------------------ Tabella aggiornata ------------------------
B: {'B': 0, 'A': 2, 'D': 3, 'C': 4, 'F': 7, 'E': 5}
--------------------------------------------------------------------
------------------------ B riceve DV(D) ------------------------
---------------------------- Tabella ----------------------------
B: {'B': 0, 'A': 2, 'D': 3, 'C': 4, 'F': 7, 'E': 5}
------------------------ Tabella aggiornata ------------------------
B: {'B': 0, 'A': 2, 'D': 3, 'C': 4, 'F': 7, 'E': 5}
--------------------------------------------------------------------
------------------------ C riceve DV(A) ------------------------
---------------------------- Tabella ----------------------------
C: {'C': 0, 'A': 2, 'D': 5, 'F': 11, 'E': 1}
------------------------ Tabella aggiornata ------------------------
C: {'C': 0, 'A': 2, 'D': 4, 'F': 7, 'E': 1, 'B': 4}
--------------------------------------------------------------------
------------------------ C riceve DV(D) ------------------------
---------------------------- Tabella ----------------------------
C: {'C': 0, 'A': 2, 'D': 4, 'F': 7, 'E': 1, 'B': 4}
------------------------ Tabella aggiornata ------------------------
C: {'C': 0, 'A': 2, 'D': 4, 'F': 7, 'E': 1, 'B': 4}
--------------------------------------------------------------------
------------------------ C riceve DV(F) ------------------------
---------------------------- Tabella ----------------------------
C: {'C': 0, 'A': 2, 'D': 4, 'F': 7, 'E': 1, 'B': 4}
------------------------ Tabella aggiornata ------------------------
C: {'C': 0, 'A': 2, 'D': 4, 'F': 7, 'E': 1, 'B': 4}
--------------------------------------------------------------------
------------------------ C riceve DV(E) ------------------------
---------------------------- Tabella ----------------------------
C: {'C': 0, 'A': 2, 'D': 4, 'F': 7, 'E': 1, 'B': 4}
------------------------ Tabella aggiornata ------------------------
C: {'C': 0, 'A': 2, 'D': 4, 'F': 7, 'E': 1, 'B': 4}
--------------------------------------------------------------------
------------------------ D riceve DV(A) ------------------------
```

```
------------------------------ Tabella ------------------------------
D: {'D': 0, 'A': 9, 'B': 3, 'C': 5, 'E': 3}
------------------------- Tabella aggiornata -------------------------
D: {'D': 0, 'A': 5, 'B': 3, 'C': 4, 'E': 3, 'F': 10}
----------------------------------------------------------------------
------------------------- D riceve DV(B) -----------------------------
------------------------------ Tabella ------------------------------
D: {'D': 0, 'A': 5, 'B': 3, 'C': 4, 'E': 3, 'F': 10}
------------------------- Tabella aggiornata -------------------------
D: {'D': 0, 'A': 5, 'B': 3, 'C': 4, 'E': 3, 'F': 10}
----------------------------------------------------------------------
------------------------- D riceve DV(C) -----------------------------
------------------------------ Tabella ------------------------------
D: {'D': 0, 'A': 5, 'B': 3, 'C': 4, 'E': 3, 'F': 10}
------------------------- Tabella aggiornata -------------------------
D: {'D': 0, 'A': 5, 'B': 3, 'C': 4, 'E': 3, 'F': 10}
----------------------------------------------------------------------
------------------------- D riceve DV(E) -----------------------------
------------------------------ Tabella ------------------------------
D: {'D': 0, 'A': 5, 'B': 3, 'C': 4, 'E': 3, 'F': 10}
------------------------- Tabella aggiornata -------------------------
D: {'D': 0, 'A': 5, 'B': 3, 'C': 4, 'E': 3, 'F': 10}
----------------------------------------------------------------------
------------------------- E riceve DV(D) -----------------------------
------------------------------ Tabella ------------------------------
E: {'E': 0, 'D': 3, 'C': 1, 'F': 7}
------------------------- Tabella aggiornata -------------------------
E: {'E': 0, 'D': 3, 'C': 1, 'F': 7, 'A': 3, 'B': 5}
----------------------------------------------------------------------
------------------------- E riceve DV(C) -----------------------------
------------------------------ Tabella ------------------------------
E: {'E': 0, 'D': 3, 'C': 1, 'F': 7, 'A': 3, 'B': 5}
------------------------- Tabella aggiornata -------------------------
E: {'E': 0, 'D': 3, 'C': 1, 'F': 7, 'A': 3, 'B': 5}
----------------------------------------------------------------------
------------------------- E riceve DV(F) -----------------------------
------------------------------ Tabella ------------------------------
E: {'E': 0, 'D': 3, 'C': 1, 'F': 7, 'A': 3, 'B': 5}
------------------------- Tabella aggiornata -------------------------
E: {'E': 0, 'D': 3, 'C': 1, 'F': 7, 'A': 3, 'B': 5}
----------------------------------------------------------------------
------------------------- F riceve DV(A) -----------------------------
------------------------------ Tabella ------------------------------
F: {'F': 0, 'A': 5, 'C': 11, 'E': 7}
------------------------- Tabella aggiornata -------------------------
F: {'F': 0, 'A': 5, 'C': 7, 'E': 7, 'B': 7, 'D': 10}
----------------------------------------------------------------------
------------------------- F riceve DV(C) -----------------------------
------------------------------ Tabella ------------------------------
F: {'F': 0, 'A': 5, 'C': 7, 'E': 7, 'B': 7, 'D': 10}
------------------------- Tabella aggiornata -------------------------
F: {'F': 0, 'A': 5, 'C': 7, 'E': 7, 'B': 7, 'D': 10}
----------------------------------------------------------------------
------------------------- F riceve DV(E) -----------------------------
```

```
----------------------------- Tabella ------------------------------
F: {'F': 0, 'A': 5, 'C': 7, 'E': 7, 'B': 7, 'D': 10}
------------------------ Tabella aggiornata ------------------------
F: {'F': 0, 'A': 5, 'C': 7, 'E': 7, 'B': 7, 'D': 10}
--------------------------------------------------------------------

Tabelle di instradamento dopo l'iterazione
Node A: A(0), B(2), C(2), D(5), E(3), F(5),
Node B: A(2), B(0), C(4), D(3), E(5), F(7),
Node C: A(2), B(4), C(0), D(4), E(1), F(7),
Node D: A(5), B(3), C(4), D(0), E(3), F(10),
Node E: A(3), B(5), C(1), D(3), E(0), F(7),
Node F: A(5), B(7), C(7), D(10), E(7), F(0),

--------------------------- Iterazione: 2 ---------------------------
------------------------- A riceve DV(B) -------------------------
------------------------------ Tabella -------------------------------
A: {'A': 0, 'B': 2, 'C': 2, 'D': 5, 'F': 5, 'E': 3}
------------------------ Tabella aggiornata ------------------------
A: {'A': 0, 'B': 2, 'C': 2, 'D': 5, 'F': 5, 'E': 3}
--------------------------------------------------------------------
------------------------- A riceve DV(C) -------------------------
------------------------------ Tabella -------------------------------
A: {'A': 0, 'B': 2, 'C': 2, 'D': 5, 'F': 5, 'E': 3}
------------------------ Tabella aggiornata ------------------------
A: {'A': 0, 'B': 2, 'C': 2, 'D': 5, 'F': 5, 'E': 3}
--------------------------------------------------------------------
------------------------- A riceve DV(D) -------------------------
------------------------------ Tabella -------------------------------
A: {'A': 0, 'B': 2, 'C': 2, 'D': 5, 'F': 5, 'E': 3}
------------------------ Tabella aggiornata ------------------------
A: {'A': 0, 'B': 2, 'C': 2, 'D': 5, 'F': 5, 'E': 3}
--------------------------------------------------------------------
------------------------- A riceve DV(F) -------------------------
------------------------------ Tabella -------------------------------
A: {'A': 0, 'B': 2, 'C': 2, 'D': 5, 'F': 5, 'E': 3}
------------------------ Tabella aggiornata ------------------------
A: {'A': 0, 'B': 2, 'C': 2, 'D': 5, 'F': 5, 'E': 3}
--------------------------------------------------------------------
------------------------- B riceve DV(A) -------------------------
------------------------------ Tabella -------------------------------
B: {'B': 0, 'A': 2, 'D': 3, 'C': 4, 'F': 7, 'E': 5}
------------------------ Tabella aggiornata ------------------------
B: {'B': 0, 'A': 2, 'D': 3, 'C': 4, 'F': 7, 'E': 5}
--------------------------------------------------------------------
------------------------- B riceve DV(D) -------------------------
------------------------------ Tabella -------------------------------
B: {'B': 0, 'A': 2, 'D': 3, 'C': 4, 'F': 7, 'E': 5}
------------------------ Tabella aggiornata ------------------------
B: {'B': 0, 'A': 2, 'D': 3, 'C': 4, 'F': 7, 'E': 5}
--------------------------------------------------------------------
------------------------- C riceve DV(A) -------------------------
------------------------------ Tabella -------------------------------
C: {'C': 0, 'A': 2, 'D': 4, 'F': 7, 'E': 1, 'B': 4}
```

```
------------------------ Tabella aggiornata ------------------------
C: {'C': 0, 'A': 2, 'D': 4, 'F': 7, 'E': 1, 'B': 4}
--------------------------------------------------------------------
------------------------- C riceve DV(D) -------------------------
----------------------------- Tabella ------------------------------
C: {'C': 0, 'A': 2, 'D': 4, 'F': 7, 'E': 1, 'B': 4}
------------------------ Tabella aggiornata ------------------------
C: {'C': 0, 'A': 2, 'D': 4, 'F': 7, 'E': 1, 'B': 4}
--------------------------------------------------------------------
------------------------- C riceve DV(F) -------------------------
----------------------------- Tabella ------------------------------
C: {'C': 0, 'A': 2, 'D': 4, 'F': 7, 'E': 1, 'B': 4}
------------------------ Tabella aggiornata ------------------------
C: {'C': 0, 'A': 2, 'D': 4, 'F': 7, 'E': 1, 'B': 4}
--------------------------------------------------------------------
------------------------- C riceve DV(E) -------------------------
----------------------------- Tabella ------------------------------
C: {'C': 0, 'A': 2, 'D': 4, 'F': 7, 'E': 1, 'B': 4}
------------------------ Tabella aggiornata ------------------------
C: {'C': 0, 'A': 2, 'D': 4, 'F': 7, 'E': 1, 'B': 4}
--------------------------------------------------------------------
------------------------- D riceve DV(A) -------------------------
----------------------------- Tabella ------------------------------
D: {'D': 0, 'A': 5, 'B': 3, 'C': 4, 'E': 3, 'F': 10}
------------------------ Tabella aggiornata ------------------------
D: {'D': 0, 'A': 5, 'B': 3, 'C': 4, 'E': 3, 'F': 10}
--------------------------------------------------------------------
------------------------- D riceve DV(B) -------------------------
----------------------------- Tabella ------------------------------
D: {'D': 0, 'A': 5, 'B': 3, 'C': 4, 'E': 3, 'F': 10}
------------------------ Tabella aggiornata ------------------------
D: {'D': 0, 'A': 5, 'B': 3, 'C': 4, 'E': 3, 'F': 10}
--------------------------------------------------------------------
------------------------- D riceve DV(C) -------------------------
----------------------------- Tabella ------------------------------
D: {'D': 0, 'A': 5, 'B': 3, 'C': 4, 'E': 3, 'F': 10}
------------------------ Tabella aggiornata ------------------------
D: {'D': 0, 'A': 5, 'B': 3, 'C': 4, 'E': 3, 'F': 10}
--------------------------------------------------------------------
------------------------- D riceve DV(E) -------------------------
----------------------------- Tabella ------------------------------
D: {'D': 0, 'A': 5, 'B': 3, 'C': 4, 'E': 3, 'F': 10}
------------------------ Tabella aggiornata ------------------------
D: {'D': 0, 'A': 5, 'B': 3, 'C': 4, 'E': 3, 'F': 10}
--------------------------------------------------------------------
------------------------- E riceve DV(D) -------------------------
----------------------------- Tabella ------------------------------
E: {'E': 0, 'D': 3, 'C': 1, 'F': 7, 'A': 3, 'B': 5}
------------------------ Tabella aggiornata ------------------------
E: {'E': 0, 'D': 3, 'C': 1, 'F': 7, 'A': 3, 'B': 5}
--------------------------------------------------------------------
------------------------- E riceve DV(C) -------------------------
----------------------------- Tabella ------------------------------
E: {'E': 0, 'D': 3, 'C': 1, 'F': 7, 'A': 3, 'B': 5}
```

```
------------------------- Tabella aggiornata ------------------------
E: {'E': 0, 'D': 3, 'C': 1, 'F': 7, 'A': 3, 'B': 5}
---------------------------------------------------------------------
-------------------------- E riceve DV(F) --------------------------
------------------------------ Tabella ------------------------------
E: {'E': 0, 'D': 3, 'C': 1, 'F': 7, 'A': 3, 'B': 5}
------------------------- Tabella aggiornata ------------------------
E: {'E': 0, 'D': 3, 'C': 1, 'F': 7, 'A': 3, 'B': 5}
---------------------------------------------------------------------
-------------------------- F riceve DV(A) --------------------------
------------------------------ Tabella ------------------------------
F: {'F': 0, 'A': 5, 'C': 7, 'E': 7, 'B': 7, 'D': 10}
------------------------- Tabella aggiornata ------------------------
F: {'F': 0, 'A': 5, 'C': 7, 'E': 7, 'B': 7, 'D': 10}
---------------------------------------------------------------------
-------------------------- F riceve DV(C) --------------------------
------------------------------ Tabella ------------------------------
F: {'F': 0, 'A': 5, 'C': 7, 'E': 7, 'B': 7, 'D': 10}
------------------------- Tabella aggiornata ------------------------
F: {'F': 0, 'A': 5, 'C': 7, 'E': 7, 'B': 7, 'D': 10}
---------------------------------------------------------------------
-------------------------- F riceve DV(E) --------------------------
------------------------------ Tabella ------------------------------
F: {'F': 0, 'A': 5, 'C': 7, 'E': 7, 'B': 7, 'D': 10}
------------------------- Tabella aggiornata ------------------------
F: {'F': 0, 'A': 5, 'C': 7, 'E': 7, 'B': 7, 'D': 10}
---------------------------------------------------------------------

Tabelle di instradamento dopo l'iterazione
Node A: A(0), B(2), C(2), D(5), E(3), F(5),
Node B: A(2), B(0), C(4), D(3), E(5), F(7),
Node C: A(2), B(4), C(0), D(4), E(1), F(7),
Node D: A(5), B(3), C(4), D(0), E(3), F(10),
Node E: A(3), B(5), C(1), D(3), E(0), F(7),
Node F: A(5), B(7), C(7), D(10), E(7), F(0),
```