

LAB 3:

Algoritmos de Enrutamiento

Descripción de la práctica

La práctica consiste en desarrollar una implementación de algoritmos de enrutamiento, con ayuda del chat desarrollado en el proyecto 1. Como objetivos de aprendizaje se reconoce el; Entender y conocer los algoritmos de enrutamiento que se utilizan actualmente en internet y saber el cómo funcionan las tablas de enrutamiento.

Los algoritmos de enrutamiento utilizados serán:

- Flooding
Es un algoritmo similar a hacer un *broadcast* en una topología de red solo que se trata de enviar un mensaje/paquete por cada nodo hasta llegar al final. Cabe recalcar que se tienen que agregar ciertas decisiones para que el algoritmo no se quede trabado en un ciclo de transmisión-recepción pues cuando llega al tope, puede que este mande de nuevo el paquete al nodo que lo envió primeramente.
- Distance Vector Routing
Primordialmente, este es un protocolo de cambio de rutas, no constante que informa al router sus vecinos directos e indirectos en la topología. Conocido históricamente como el algoritmo de ARPANET. Este algoritmo se basa en la tabla de enrutamiento de Bellman Ford. Esta tabla de enrutamiento se establece para cada nodo, y solo se sabrán los pesos de sus vecinos directos, mientras que los demás vecinos de la topología se irán calculando, conforme se necesite enviar un mensaje hasta un punto donde cada nodo conozca las rutas y pesos hacia todos los nodos. Se apoya de esta ecuación para determinar las rutas:

Ecuación Bellman-Ford

Sea $D_x(y)$: costo de la ruta de menor costo de x a y .
Entonces:

$$D_x(y) = \min_v \{ c_{x,v} + D_v(y) \}$$

\min tomado sobre todos los
vecinos v de x

$c_{x,v}$ costo directo del enlace de x a v
 $D_v(y)$ Ruta de menor costo estimada de v 's a y

- Link State Routing

Este algoritmo se utiliza en el caso en que routers conocen las tablas de enrutamiento usando la misma lógica que Dijkstra, el cual es encontrar la ruta más corta entre dos nodos. En este caso como los pesos son 1, entonces lo que será la ruta corta se basará en el número de “saltos” posibles.

Resultados

```
≡ topologia.txt
1  {
2  "type":"topo",
3  "config":{
4    "A": ["B", "D", "F"],
5    "B": ["A", "C", "D", "F"],
6    "C": ["B", "E", "F"],
7    "D": ["A", "B"],
8    "E": ["C", "F"],
9    "F": ["A", "B", "C", "E"]}
10 }
```

Imagen 1: Topología para pruebas

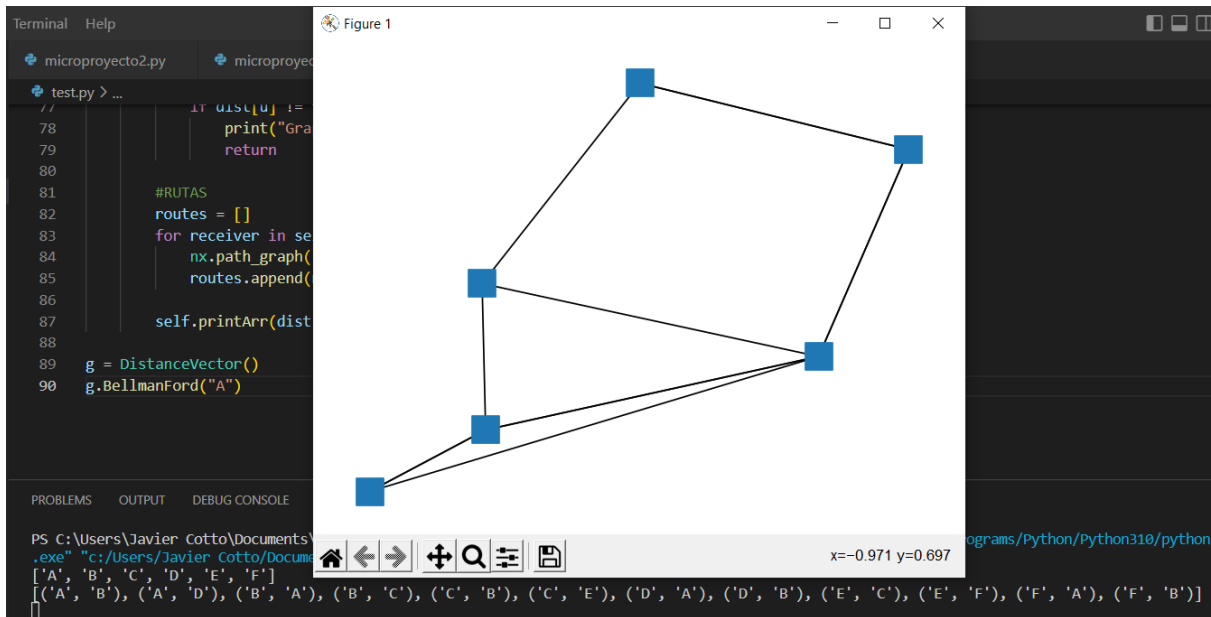


Imagen 2: Grafo mostrado a partir de la topología de prueba

Distance Vector

VERTICES	PESO	RUTA
A	0	['A']
B	1	['A', 'B']
C	2	['A', 'B', 'C']
D	1	['A', 'D']
E	3	['A', 'B', 'C', 'E']
F	4	['A', 'B', 'C', 'E', 'F']

Imagen 3: Tabla de enrutamiento para el nodo A, con sus respectivas rutas más cortas hacia todos los nodos.

Flooding

```

-----
| MENU DE OPCIONES |
| 1. Iniciar Sesión |
| 2. Salir          |
|-----|
Ingrese opción: 1
Ingrese usuario(usuario@alumchat.fun): walt.gfe@alumchat.fun
Ingrese contraseña:
1) Flooding
2) Distance Vector
3) Dijkstra
Ingrese algoritmo: 1
['A', 'B', 'C', 'D', 'E', 'F']
[('A', 'B'), ('A', 'D'), ('B', 'A'), ('B', 'C'), ('C', 'B'), ('C', 'E'), ('D', 'A'), ('D', 'B'), ('E', 'C'), ('E', 'F'), ('F', 'A'), ('F', 'B')]
-----
| MENU DE FUNCIONES |
| 1. Mostrar Contactos |
| 2. Agregar Contacto |
| 3. Detalles Contacto |
| 4. Mensaje Directo |
| 5. Conversación Grupal |
| 6. Mensaje de Presencia |
| 7. Archivos |
| 8. Eliminar |
| 9. Salir |
| 10. Obtener ruta |
|-----|
Ingrese opción: 10
Nodo inicial: A
Nodo receptor: B
{'A': ['B', 'D'], 'B': ['C'], 'D': [], 'C': ['E'], 'E': ['F'], 'F': []}

```

Imagen 4: Tabla de enrutamiento para el nodo A para enviar a todos los nodos con flooding.

```

-----
| MENU DE OPCIONES |
-----
1. Iniciar Sesión
2. Salir
-----
Ingrese opción: 1
Ingrese usuario(usuario@alumchat.fun): walt.gfe@alumchat.fun
Ingrese contraseña:
1) Flooding
2) Distance Vector
3) Dijkstra
Ingrese algoritmo: 3
['A', 'B', 'C', 'D', 'E', 'F']
[['A', 'B'], ['A', 'D'], ['B', 'A'], ['B', 'C'], ['C', 'B'], ['C', 'E'], ['D', 'A'], ['D', 'B'], ['E', 'C'], ['E', 'F'], ['F', 'A'], ['F', 'B']]
-----
| MENU DE FUNCIONES |
-----
1. Mostrar Contactos
2. Agregar Contacto
3. Detalles Contacto
4. Mensaje Directo
5. Conversación Grupal
6. Mensaje de Presencia
7. Archivos
8. Eliminar
9. Salir
10. Obtener ruta
-----
Ingrese opción: 10
Nodo inicial: A
Nodo receptor: C
{'A': [['A'], ['A', 'B'], ['A', 'B', 'C'], ['A', 'D'], ['A', 'F', 'E'], ['A', 'F']], 'B': [['B', 'A'], ['B', 'B', 'C'], ['B', 'D'], ['B', 'C', 'E'], ['B', 'F']], 'C': [['C', 'B', 'A'], ['C', 'B'], ['C'], ['C', 'B', 'D'], ['C', 'E'], ['C', 'B', 'F']], 'D': [['D', 'A'], ['D', 'B'], ['D', 'B', 'C'], ['D'], ['D', 'B', 'C', 'E'], ['D', 'A', 'F']], 'E': [['E', 'F', 'A'], ['E', 'C', 'B'], ['E', 'C'], ['E', 'F', 'A', 'D'], ['E'], ['E', 'F']], 'F': [['F', 'A'], ['F', 'B'], ['F', 'B', 'C'], ['F', 'A', 'D'], ['F', 'E'], ['F']]}
['A', 'B', 'C']

```

Discusión

Para nuestro laboratorio se implementaron los algoritmos para calcular todas las rutas posibles. Luego a partir de las tablas de enrutamiento, nosotros manualmente enviaremos un mensaje a los otros nodos. Tuvimos un problema, pues el recibir mensajes y reenviarlos automáticamente se logró implementar y no, pues las líneas de código si se encuentran en el programa solo que el código o implementación decide omitirlas y pues no realizan nada.

Comentario Grupal

Una práctica interesante, se pudo ver en realidad cómo interactúa el internet con los dispositivos actualmente y como las propias tipologías tienen que estar propensas a errores para que no suceda ningún imprevisto. El haber implementado y actualizado las topologías de internet con el pasar del tiempo, tuvo que llevar a un uso de bastantes recursos, para lograr lo que hemos logrado. Luego de esto como camino a futuro, se puede ver que las topologías seguramente se reducirán a cada nodo ubicado en la nube y ya no físico.

Conclusiones

- *Link State Routing* y *Distance Vector* son algoritmos similares en su funcionamiento, pues los dos calculan la ruta más corta. Donde difieren, es que en LSR, cada nodo sabe la topología completa y procede a calcular todas las rutas desde un inicio, mientras que DV solo calcula las rutas, en caso las necesite de lo contrario la tabla de ruteo de cada nodo, seguirá siendo igual.