

EJERCICIO 2:

a) Al ejecutar su código, ¿observa alguna variación en el número de nodos alcanzados entre diferentes ejecuciones? Explique el motivo si esto sucediese.

Al ejecutar el código varias veces, vemos que hay variaciones en el número de nodos alcanzados. Esto es porque el algoritmo de propagación del mensaje es aleatorio porque como vemos en el código:

1. La probabilidad de detención es decir el p-stop es aleatorio es decir q el nodo decide si retransmitir o no cuando lo ejecutamos.
2. Al retransmitir el mensaje el nodo selecciona aleatoriamente sus vecinos, entonces diferentes elecciones son diferentes rutas de propagación y así un numero aleatorio de nodos.
3. Hay nodos que pueden alcanzar más nodos por su propagación dependiendo de la red.
4. Aunque todas las ejecuciones comienzan desde el mismo estado inicial, es decir el 1, las decisiones aleatorias de los nodos pueden dar lugar a diferentes comportamientos en cada ejecución.

```
PS C:\Users\marin\OneDrive\Desktop\PEC1_SD_Marina_Amaya\Ejercicio2> python ejercicio2.py
Node 1 received: Important update!
Node 4 received: Important update!
Node 4 ha decidido no retransmitir el rumor (p_stop = 0.2)
Node 2 received: Important update!
Node 2 ha decidido no retransmitir el rumor (p_stop = 0.2)
Node 3 received: Important update!
Node 7 received: Important update!
Node 8 received: Important update!
Node 14 received: Important update!
Node 15 received: Important update!

Difusión completa.
El mensaje ha llegado a 8 nodos de 20.
PS C:\Users\marin\OneDrive\Desktop\PEC1_SD_Marina_Amaya\Ejercicio2> python ejercicio2.py
Node 1 received: Important update!
Node 2 received: Important update!
Node 4 received: Important update!
Node 3 received: Important update!
Node 6 received: Important update!
Node 6 ha decidido no retransmitir el rumor (p_stop = 0.2)
Node 5 received: Important update!
Node 5 ha decidido no retransmitir el rumor (p_stop = 0.2)
Node 9 received: Important update!
Node 10 received: Important update!
Node 7 received: Important update!
Node 7 ha decidido no retransmitir el rumor (p_stop = 0.2)
Node 8 received: Important update!
Node 15 received: Important update!

Difusión completa.
El mensaje ha llegado a 11 nodos de 20.
```

```

PS C:\Users\marin\OneDrive\Desktop\PEC1_SD_Marina_Amaya\Ejercicio2> python ejercicio2.py
Node 1 received: Important update!
Node 4 received: Important update!
Node 4 ha decidido no retransmitir el rumor (p_stop = 0.2)
Node 3 received: Important update!
Node 3 ha decidido no retransmitir el rumor (p_stop = 0.2)
Node 2 received: Important update!
Node 2 ha decidido no retransmitir el rumor (p_stop = 0.2)

Difusión completa.
El mensaje ha llegado a 4 nodos de 20.
PS C:\Users\marin\OneDrive\Desktop\PEC1_SD_Marina_Amaya\Ejercicio2> python ejercicio2.py
Node 1 received: Important update!
Node 4 received: Important update!
Node 3 received: Important update!
Node 3 ha decidido no retransmitir el rumor (p_stop = 0.2)
Node 2 received: Important update!
Node 10 received: Important update!
Node 9 received: Important update!
Node 5 received: Important update!
Node 6 received: Important update!
Node 11 received: Important update!
Node 12 received: Important update!
Node 13 received: Important update!
Node 13 ha decidido no retransmitir el rumor (p_stop = 0.2)

Difusión completa.
El mensaje ha llegado a 11 nodos de 20.

```

b) ¿Afecta el valor de p-stop al número de nodos que reciben el mensaje? ¿Por qué?

Sí, afecta porque para tener una idea el **p-stop** es la probabilidad de que cada nodo utiliza para decir si retransmite el mensaje o no, esto lo hace generando un número aleatorio entre 0 y 1 y si este número es menor que el p-stop el nodo no retransmite el mensaje.

Una vez sabemos esto **¿Cómo afecta el valor del p-stop al número de nodos que reciben el mensaje?**

- Un valor alto del p-stop significa que hay una mayor probabilidad de que el nodo decida detener la propagación, si muchos nodos deciden no retransmitir el mensaje debido a el p-stop alto la propagación se detiene más rápido y por lo tanto se alcanza un número menor nodos totales. Al contrario de lo que pasa en nuestro programa que tenemos un p-stop de 0.2 que cada nodo tiene un 20% de probabilidad de que no retransmita el mensaje, esto significa que es más probable que el nodo siga retransmitiendo el mensaje y por lo tanto resulta en que más nodos reciban el mensaje, lo que aumenta el número de nodos alcanzados.