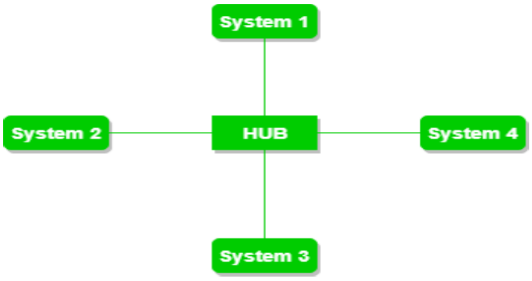
EXPERIMENT-7

AIM: Configure and study a network using star topology.

THEORY: The arrangement of a network that comprises nodes and connecting lines via sender and receiver is referred to as network topology.
Star topology: In star topology, all the devices are connected to a single hub through a cable. This hub is the central node and all other nodes are connected to the central node. The hub can be passive in nature i.e., not an intelligent hub such as broadcasting devices, at the same time the hub can be intelligent known as an active hub. Active hubs have repeaters in them.



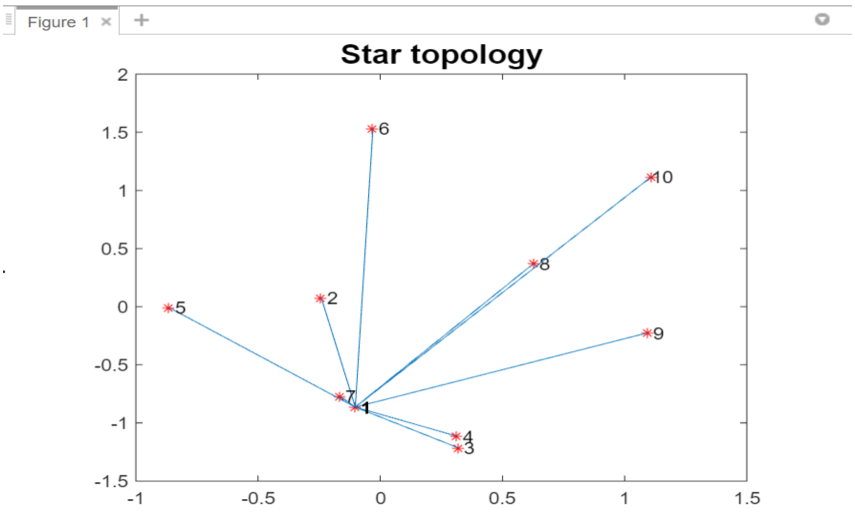
CODE:

```
clc;
clear;
x = randn(10, 1);
y = randn(10, 1);
plot(x, y,'r*');
hold on;

for center = 1
    for clients = 2 : 10
        line([x(center, 1), x(clients, 1)], [y(center, 1), y(clients,1)])
        text(x(center, 1), y(center, 1), sprintf('%2.0f', center))
        text(x(clients, 1), y(clients, 1), sprintf('%2.0f', clients))

    title('Star topology','fontsize', 16)
    end
end
```

OUTPUT:



EXPERIMENT-8

AIM: Configure and study a network using ring topology.

THEORY: A **ring topology** is a network configuration where device connections create a circular data path. Each networked device is connected to two others, like points on a circle. Together, devices in a ring topology are referred to as a **ring network**.

In a ring network, packets of data travel from one device to the next until they reach their destination. Most ring topologies allow packets to travel only in one direction, called a **unidirectional** ring network. Others permit data to move in either direction, called **bidirectional**.

The major disadvantage of a ring topology is that if any individual connection in the ring is broken, the entire network is affected.

Ring topologies may be used in either local area networks or wide area networks.

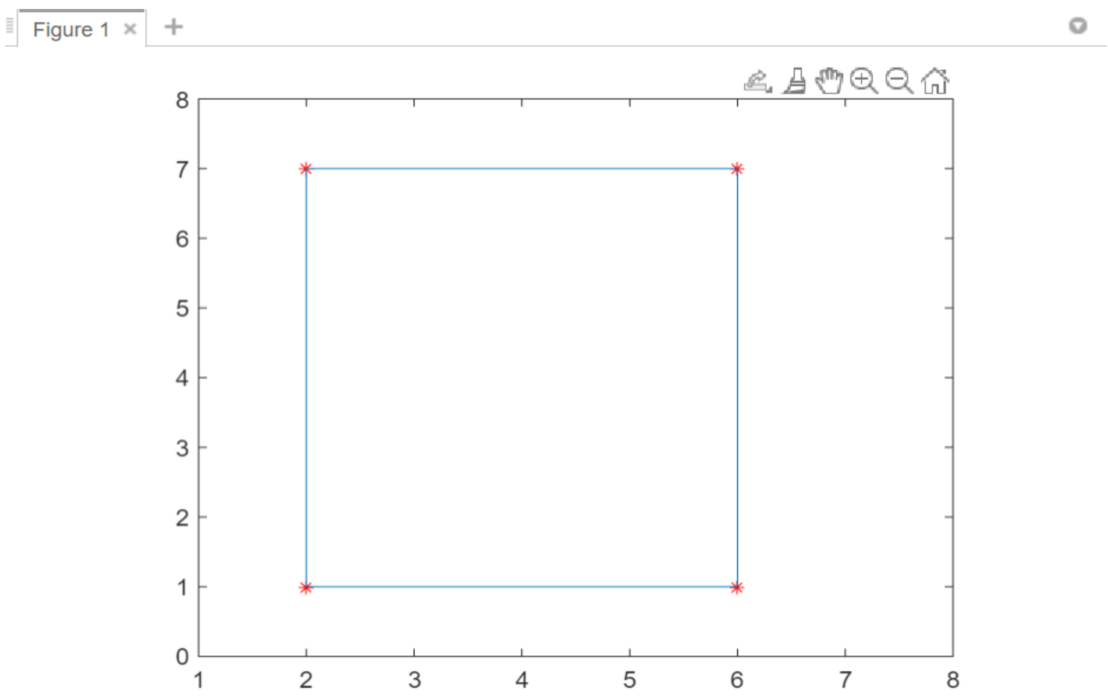
CODE:

```
clc;
clear;
nodesXcoordinates = [6, 2, 2, 6];
nodexYcoordinates = [7, 7, 1, 1];
figure(1);
plot(nodesXcoordinates, nodexYcoordinates,'r*');
xlim([1 8]);
ylim([0 8]);

hold on;

line([nodesXcoordinates(1), nodesXcoordinates(2)],[nodexYcoordinates(1), nodexYcoordinates(2)]);
line([nodesXcoordinates(2), nodesXcoordinates(3)],[nodexYcoordinates(2), nodexYcoordinates(3)]);
line([nodesXcoordinates(3), nodesXcoordinates(4)],[nodexYcoordinates(3), nodexYcoordinates(4)]);
line([nodesXcoordinates(4), nodesXcoordinates(1)],[nodexYcoordinates(4), nodexYcoordinates(1)]);
```

OUTPUT:



CODE FOR BER:

```
clear;

N = 10^6;
a = rand(1, N) > 0.5;
s = 2*a - 1;
snr_dB = 1 : 1 : 10;
snr_ratio = 10.^snr_dB / 10;
n = 1 / sqrt(2).*(randn(1, N) + 1i*randn(1, N));
for i = 1 : length(snr_dB)
    y = 10^(snr_dB(i) / 20).*s + n;
    a_dec = real(y) > 0;
    nErr(i) = size(find(a- a_dec),2);
end
simBer = nErr/N;
theoryBer = 0.5*erfc(sqrt(10.^(snr_dB/10)));
figure
semilogy(snr_dB,theoryBer,'b-','Linewidth',1.5);
hold on
semilogy(snr_dB,simBer,'x','MarkerSize',8);
grid on
legend('theory','simulation');
xlabel('snr dB-->');
ylabel('Bit Error Rate-->');
title('Bit error probability curve for BPSK modulation');
```

OUTPUT OF BER: