-:COMPUTER NETWORKS LAB 8-:

~Pranay Karvi 23BDS1137

CODE-: C

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <unistd.h>

#define MAX_WAIT_TIME 5          // Maximum wait time in seconds
#define MAX_TRANSMISSION_TIME 5   // Maximum time to transmit a packet
#define COLLISION_PROBABILITY 0.3 // Probability of collision occurring

// Function to simulate a random delay before transmission
void random_wait(int max_wait_time) {
   int wait_time = rand() % max_wait_time + 1;
   printf("Waiting for %d seconds...\n", wait_time);
   sleep(wait_time);
}

// Function to simulate a packet transmission
int transmit_packet() {
   int transmission_time = rand() % MAX_TRANSMISSION_TIME + 1;
   printf("Transmitting packet for %d seconds...\n", transmission_time);
   sleep(transmission_time);
```

```
    // Simulate collision probability
    float collision_chance = (float)rand() / (float)RAND_MAX;
    if (collision_chance < COLLISION_PROBABILITY) {
        return 1; // Collision occurred
    }
    return 0; // No collision
}

// CSMA/CD simulation for a single node
void csma_cd_node(int node_id) {
    int attempts = 0;
    int max_attempts = 5;

    while (attempts < max_attempts) {
        printf("Node %d - Attempt #%d\n", node_id, attempts + 1);

        // Sense the channel (check if it's busy)
        printf("Sensing the channel...\n");

        // Randomly decide if the channel is busy or free (for simulation purposes)
        int channel_busy = rand() % 2; // 0 for free, 1 for busy
        if (channel_busy) {
            printf("Channel is busy, waiting...\n");
            random_wait(MAX_WAIT_TIME);
            attempts++;
```

```
        } else {
            printf("Channel is free, starting transmission...\n");
            int collision = transmit_packet();

            if (collision) {
                printf("Collision detected! Retrying...\n");
                attempts++;
                random_wait(MAX_WAIT_TIME);  // Wait before retrying
            } else {
                printf("Node %d - Packet transmitted successfully!\n", node_id);
                break;
            }
        }
    }

    if (attempts >= max_attempts) {
        printf("Node %d - Failed to transmit after %d attempts.\n", node_id, max_attempts);
    }
}

int main() {
    int mnodes;

    // Seed the random number generator
    srand(time(NULL));
```

```
    // Take user input for the number of nodes
    printf("Enter the number of nodes: ");
    scanf("%d", &mnodes);

    // Simulate CSMA/CD for each node
    for (int i = 1; i <= mnodes; i++) {
        printf("\n\nSimulating CSMA/CD for Node %d:\n", i);
        csma_cd_node(i);
    }

    return 0;
}
```

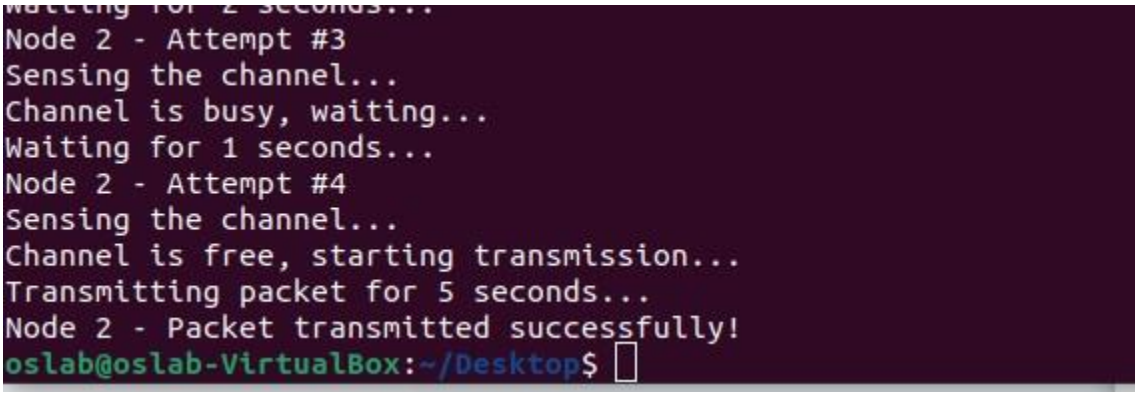
SCREENSHOTS-:

```
oslab@oslab-VirtualBox:~/Desktop$ gcc csma-cd.c -o csma
oslab@oslab-VirtualBox:~/Desktop$ ./csma
Enter the number of nodes: 2


Simulating CSMA/CD for Node 1:
Node 1 - Attempt #1
Sensing the channel...
Channel is busy, waiting...
Waiting for 2 seconds...
Node 1 - Attempt #2
Sensing the channel...
Channel is busy, waiting...
Waiting for 4 seconds...
Node 1 - Attempt #3
Sensing the channel...
Channel is busy, waiting...
Waiting for 3 seconds...
Node 1 - Attempt #4
Sensing the channel...
Channel is busy, waiting...
Waiting for 1 seconds...
```

```
Node 1 - Attempt #5
Sensing the channel...
Channel is free, starting transmission...
Transmitting packet for 3 seconds...
Collision detected! Retrying...
Waiting for 5 seconds...
Node 1 - Failed to transmit after 5 attempts.


Simulating CSMA/CD for Node 2:
Node 2 - Attempt #1
Sensing the channel...
Channel is free, starting transmission...
Transmitting packet for 4 seconds...
Collision detected! Retrying...
Waiting for 3 seconds...
Node 2 - Attempt #2
Sensing the channel...
Channel is busy, waiting...
Waiting for 2 seconds...
Node 2 - Attempt #3
Sensing the channel...
Channel is busy, waiting...
Waiting for 1 seconds...
```


Node 2 - Attempt #3
Sensing the channel...
Channel is busy, waiting...
Waiting for 1 seconds...
Node 2 - Attempt #4
Sensing the channel...
Channel is free, starting transmission...
Transmitting packet for 5 seconds...
Node 2 - Packet transmitted successfully!
oslab@oslab-VirtualBox:~/Desktop$