# Project 2 Report

Name: Sayed Erfan Arefin
R#11659978

## Instructions to run the code

The code can be compiled using the following command:

```
csim64.gcc pro2_Arefin_SayedErfan.c -o pro2_Arefin_SayedErfan
```

It produces some warning messages but no errors during compilation.
After compilation, it can be executed using the following command:

```
./pro2_Arefin_SayedErfan 0.3
```

The program takes in 1 parameter. This is the packet loss probability. It needs to be between in this range (0< packet loss probability < 1).
The provided packet loss probability is saved in a global variable and is used during the simulation.

## Two result graphs

Two result graphs in terms of following performance metrics against the packet loss probabilities (0.1, 0.2, 0.3, 0.4, or 0.5).

1. Average number of successful transmissions
2. Average number of failed transmissions

This graph is provided in Figure 1.

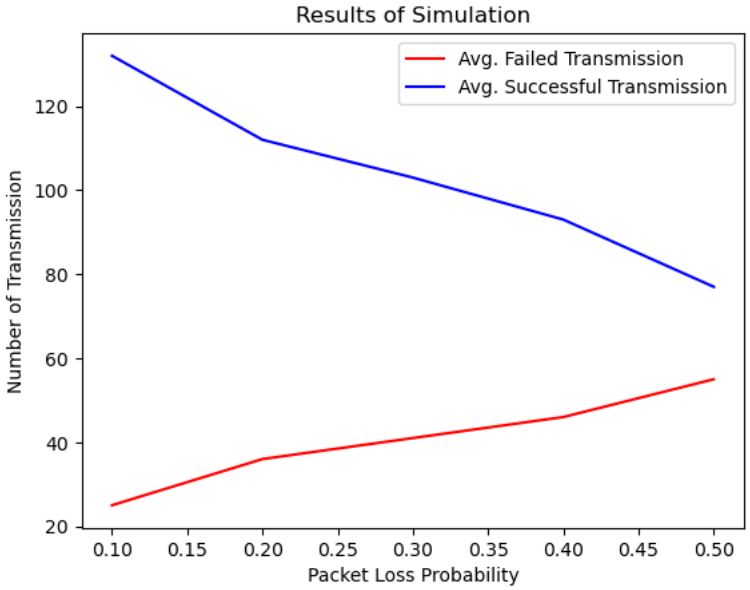


*Figure 1 Performance graph for different Packet loss probabilities (0.1, 0.2, 0.3, 0.4, 0.5)*

## Snapshot of events dump for packet loss probability 0.3

```
cpu 0cpu 1cpu 2cpu 3cpu 4
node. 4: sends hello to node.1 at  0.269 seconds
node. 4: re sends hello to node.1 at  2.569 seconds
node. 3: sends hello to node.2 at  3.132 seconds
node. 0: sends hello to node.1 at  3.329 seconds
node. 3: re sends hello to node.2 at  5.432 seconds
node. 0: re sends hello to node.1 at  5.629 seconds
node. 2: sends hello to node.3 at  5.878 seconds
node. 2: replies a hello_ack to node.3 at  6.178 seconds
node. 3: sends hello to node.4 at  7.503 seconds
node. 1: sends hello to node.2 at  8.694 seconds
node. 1: replies a hello_ack to node.4 at  8.994 seconds
node. 2: sends hello to node.3 at 10.824 seconds
node. 2: replies a hello_ack to node.1 at 11.124 seconds
node. 4: sends hello to node.3 at 11.284 seconds
node. 4: replies a hello_ack to node.3 at 11.584 seconds
node. 4: receives a hello_ack to node.3 at 11.884 seconds
node. 1: sends hello to node.2 at 12.566 seconds
node. 1: replies a hello_ack to node.0 at 12.866 seconds
node. 1: receives a hello_ack to node.0 at 13.166 seconds
node. 3: sends hello to node.4 at 14.370 seconds
node. 3: replies a hello_ack to node.2 at 14.670 seconds
node. 3: sends hello to node.2 at 15.182 seconds
node. 4: sends hello to node.1 at 16.601 seconds
node. 2: sends hello to node.4 at 17.268 seconds
node. 0: sends hello to node.3 at 17.351 seconds
node. 1: sends hello to node.4 at 17.552 seconds
node. 4: replies a hello_ack to node.1 at 17.852 seconds
node. 3: sends hello to node.2 at 18.226 seconds
node. 1: sends hello to node.4 at 20.643 seconds
node. 4: sends hello to node.2 at 23.854 seconds
node. 4: replies a hello_ack to node.1 at 24.154 seconds
node. 3: sends hello to node.1 at 25.032 seconds
node. 0: sends hello to node.1 at 27.155 seconds
node. 3: re sends hello to node.1 at 27.332 seconds
node. 4: sends hello to node.2 at 27.738 seconds
node. 1: sends hello to node.4 at 28.076 seconds
node. 0: re sends hello to node.1 at 29.455 seconds
node. 4: re sends hello to node.2 at 30.038 seconds
node. 1: sends hello to node.4 at 30.521 seconds
node. 1: replies a hello_ack to node.0 at 30.821 seconds
node. 1: receives a hello_ack to node.0 at 31.121 seconds
node. 1: sends hello to node.2 at 31.288 seconds
node. 3: sends hello to node.4 at 31.588 seconds
node. 2: sends hello to node.1 at 31.689 seconds
node. 2: replies a hello_ack to node.3 at 31.989 seconds
node. 1: replies a hello_ack to node.2 at 31.989 seconds
node. 0: sends hello to node.4 at 32.372 seconds
node. 0: re sends hello to node.4 at 34.672 seconds
node. 3: sends hello to node.4 at 35.962 seconds
node. 1: sends hello to node.4 at 37.161 seconds
node. 4: sends hello to node.2 at 38.449 seconds
node. 3: sends hello to node.1 at 40.633 seconds
node. 0: sends hello to node.1 at 41.185 seconds
node. 1: sends hello to node.4 at 42.111 seconds
node. 2: sends hello to node.4 at 42.121 seconds
node. 2: replies a hello_ack to node.4 at 42.421 seconds
node. 2: receives a hello_ack to node.4 at 42.721 seconds
node. 4: sends hello to node.3 at 42.957 seconds
node. 4: receives a hello_ack to node.2 at 43.433 seconds
```

```
node. 4: receives a hello_ack to node.2 at 43.433 seconds
node. 4: sends hello to node.3 at 46.035 seconds
node. 0: sends hello to node.4 at 46.730 seconds
```