**PSG COLLEGE OF TECHNOLOGY**

**DEPARTMENT OF APPLIED MATHEMATICS AND COMPUTATIONAL SCIENCES**

**COMPUTER NETWORKS LAB - PROBLEM SHEET**

**BYTE STUFFING, BIT STUFFING AND FRAMING**

**Develop a client-server program in python to perform the Data link layer tasks of byte stuffing, bit stuffing and framing.**

**Client Operation**

1. Get a message as input from the user at the client side.

Example: A ESC B ESC FLAG

1. Perform byte stuffing.

After performing byte stuffing the message becomes

A **ESC** ESC B **ESC** ESC **ESC** FLAG

1. Convert it into binary format.

The value of FLAG: 01111110; ESC: 10100011

The ASCII values of different characters are given at the end of the document.

So the example message A ESC ESC B ESC ESC ESC FLAG becomes

01000001 10100011 10100011 01000010 10100011 10100011 10100011 01111110

1. Perform bit stuffing – stuffing 0 after 0 followed by 5 consecutive 1’s.

01000001 10100011 10100011 01000010 10100011 10100011 10100011 011111**0**10

1. Perform fixed size or variable-size framing – chose frame size of your choice. Then add FLAG to start and end of each frame and sent it to the server one by one.

Suppose assume 16 data bits can be sent in one frame, Then the message can be divided into 4 frames with FLAG value to denote start and end of each frame as follows:

**01111110** 01000001 10100011 **01111110**

**01111110** 10100011 01000010 **01111110**

**01111110** 10100011 10100011 **01111110**

**01111110** 10100011 011111010 **01111110**

**Server Operation**

1. Receive the frames
2. Remove the start and end FLAG values of each frame.
3. Perform bit unstuffing and byte unstuffing
4. Generate the original message.

**ASCII Code – Character to binary is given for your reference**

