Green University of Bangladesh
Department of Computer Science and Engineering (CSE)

Faculty of Sciences and Engineering

Semester: (Fall, Year:2021), B.Sc. in CSE (Day)

LAB REPORT NO #01

Course Title: Data Communication Lab

Course Code:308 Section:193PC/DA

Lab Experiment Name: BitStuffing and BitDestuffing

Student Details

	Name	ID
1.	Mohammad Nazmul Hossain	193902031
2.		
3.		

Lab Date : 4 - 12 - 21

Submission Date : 6 - 12 - 21

Course Teacher's Name : Amena Zahan

[For Teachers use only: Don't Write Anything inside this box]

Lab Report Status	
Marks: ……	**Signature:.....**
Comments:.....	**Date:.....**

1. BitStuffing and BitdeStuffing

2. OBJECTIVES/AIM [1]

Perform on BitStuffing and BitDestuffing and implement on a programming language like **c/c++.**

3. PROCEDURE / ANALYSIS / DESIGN [2]

Algorithm for Bit−Stuffing/DeStuffing

1. Start
2. Initialize the array for the transmitted stream with the special bit pattern 0111 1110
3. which indicates the beginning of the frame.
4. Get the bitstream to be transmitted into the array.
5. Check for five consecutive ones and if they occur, stuff a bit 0
6. Display the data transmitted as it appears on the data line after appending 0111 1110 at the end
7. For de−stuffing, copy the transmitted data to another array after detecting the stuffed bits
8. Display the received bitstream
9. Stop

4. IMPLEMENTATION [2]

1 - BitStuffing in C++

Code - BitStuffing in C++

```cpp
//Bit stuffing
#include <iostream>
#include <string>
using namespace std;
```

```
/*
* Student name: Mohammad Nazmul Hossain
* Student id: 193902031
* Student Sections: PC-DA
*/
int main()
{
   string stream, stuffedStream;
   cout << "Enter the stream of bits:";
   cin >> stream;
   stuffedStream = stream;

   int count = 0, j,  appendedBit = 0;
   int l = stream.length();
   for (int i = 0; i < stream.length(); i++)
   {
       if (stream[i] == '1')
           count++;
       if (stream[i] == '0')
           count = 0;
       if (count == 5)
       {
           count = 0;
           stuffedStream += "0";
           l++;
           appendedBit++;
           j = l;
           while (j > i + 2)
           {
               stuffedStream[j - 1] = stuffedStream[j - 2];
               j--;
           }
           stuffedStream[i + appendedBit] = '0';
```

```
        }
    }
    cout << "Output stream after bit stuffing: "
         << "01111110 " << stuffedStream << " 01111110" << endl;
    return 0;
}
```

2 - BitDeStuffing in C++

Code - BitDeStuffing in C++

```
#include <iostream>
#include <string>
#define MAXSIZE 100

/*
* Student name: Mohammad Nazmul Hossain
* Student id: 193902031
* Student Sections: PC-DA
*/
using namespace std;
int main()
{
    char *p, *q;
    char temp;
    char in[MAXSIZE];
    char stuff[MAXSIZE];
    char destuff[MAXSIZE];
    int count(0);
    cout << "Enter input string(0's & 1's only):" << endl;
    cin >> in;
```

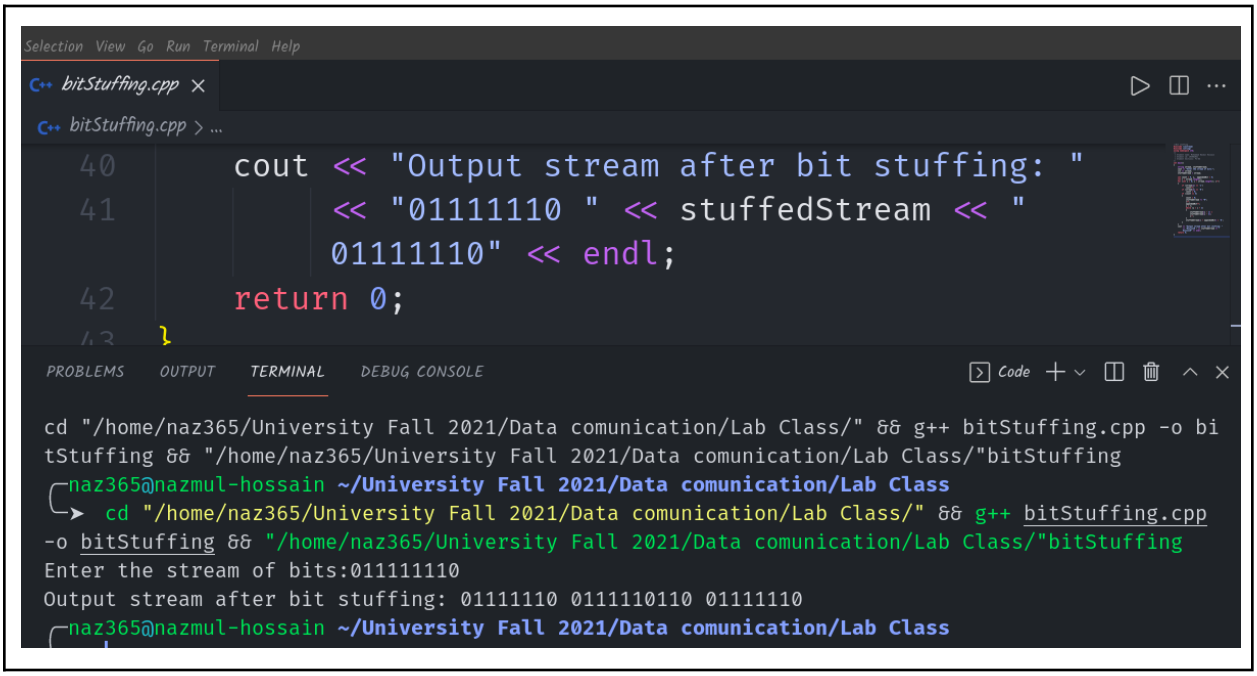
```
   //stuffing on sender's
site.....................................
   cout << "\n ——— After BitStuffing ———" << endl;
   p = in;
   q = stuff;

   while (*p != '\0')
   {
       if (*p == '0')
       {
           *q = *p;
           q++;
           p++;
       }
       else
       {
           while (*p == '1' && count != 5)
           {
               count++;
               *q = *p;
               q++;
               p++;
           }
           if (count == 5)
           {
               *q = '0';
               q++;
           }
           count = 0;
       }
   }
   *q = '\0';
   cout << "\n BitStuffed character string : " << stuff << endl;
```

```
   //destuffing on reciever's
site.....................................................
   cout << "\n ============== After BitDeStuffing
==============" << endl;
   p = stuff;
   q = destuff;
   while (*p == '\0')
   {
       if (*p == '0')
       {
           *q = *p;
           q++;
           p++;
       }
       else
       {
           while (*p == '1' && count != 5)
           {
               count++;
               *q = *p;
               q++;
               p++;
           }
           if (count == 5)
           {
               p++;
           }
           count++;
       }
   }
   *q = '\0';
   cout << "\n BitDeStuffed character string :";
   cout << in;
```

```
}
```

5. TEST RESULT / OUTPUT [2]

Output - BitStuffing in C++



Output - BitDeStuffing in c++