EAST WEST UNIVERSITY

Department of CSE
Lab Report

Course Code and Name: CSE350 – Data communication	
Lab: 02 Print every 125-character segment individually from receiver	
Name of Student: Intesar Islam Khan ID: 2019-1-60-043	**Course Instructor information:** **Md. Mahir Ashhab** Lecturer Department of Computer Science & Engineering
Section: 02	**Date of Submission:** 09/11/2022

Source code: https://github.com/IntesarEWU/CSE350-Data-communication-LAB/tree/main/Lab2

Introduction:

The OSI model is composed of seven ordered layers: 1. Application, 2. Presentation, 3. Session, 4. Transport, 5. Network, 6. Data Link, 7. Physical.

1. **Application layer:** in this layer system add extra 3 chars (`A-H`) to input file or text. One the other hand in receiver side the extra 3 chars are removed.
2. **Presentation layer:** In sender side more 3 chars (`P-H`) is added with the text or file. One the other side this extra 3 chars are removed.
3. **Session layer:** Similarly in **Sender** side more 3 chars (`S-H`) is added with the text or file. One the other side in **Receiver** this extra 3 chars are removed.
4. **Transport layer:** Similarly in **Sender** side more 3 chars (`T-H`) is added with the text or file. One the other side in **Receiver** this extra 3 chars are removed.
5. **Network layer:** Similarly in **Sender** side more 3 chars (`N-H`) is added with the text or file. One the other side in **Receiver** this extra 3 chars are removed.
6. **Data Link layer:** Similarly in **Sender** side more 6 chars (`D-H` in the beginning and `D-T` in the end) is added with the text or file. One the other side in **Receiver** this extra 6 chars are removed.
7. **Physical layer:** Similarly in **Sender** side more 4 chars (`PH-H`) is added with the text or file. One the other side in **Receiver** this extra 4 chars are removed.

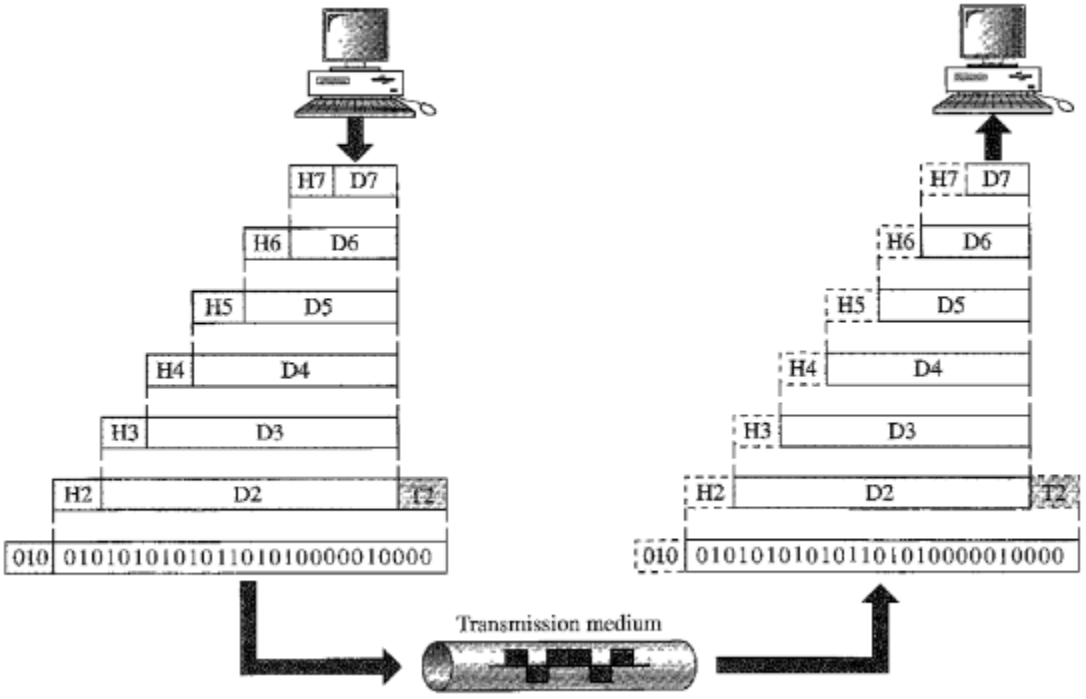

Figure 1 Sender (Left) Receiver (Right)

Lab Task:

Creating a sender receiver environment to show how OSI model actually works and print every 125-character segment from receiver side

Implementation:

I created 3 separated c++ files 1. main.cpp, 2. sender.cpp, 3. seceiver.cpp and also 3 text files 1. input.txt, 2. temp.txt, 3. output.txt.

main.cpp:

main contains the call function and a removes every older data from text files (temp.txt, output.txt)

```cpp
#include <iostream>
#include "sender.cpp"
#include "receiver.cpp"
using namespace std;

int main()
{
    ofstream temp;
    temp.open("temp.txt");
    temp << "";
    temp.close();
    ofstream output;
    output.open("output.txt");
    output << "";
    output.close();
    sender();
    receiver();

    cout<<"\n\n";
    return 0;
}
```

sender.cpp:

to create the OSI environment I create 7 more function which works as layers.

sender (function): reads text file or takes user input and send the string to application layer.

```cpp
void sender()
{
    string txt;
    printf("\nSender  : ");
    getline(cin,txt);
    //ifstream input("input.txt");
    /*while(getline(input,txt))
    {
            //cout << str << '\n';
    }*/
    int txtlen = txt.size();
    //cout << txtlen <<'\n';
    int i=0;
    while(i!=(txtlen/125)+1)
    {
        string str = txt.substr(125*i,125);
        //cout << str << '\n';
        applicationLayer(str);
        i++;
    }
}
```

Application layer(function): add 3 chars (in this case `A-H`) with the string as head and send the new string to presentation function.

```
62  void applicationLayer(string strl)
63  {
64      //char str[100]="A-H";
65      string str = "A-H"+strl;
66      //printf("%s\n",str);
67      //cout << str <<"\n";
68      presentationLayer(str);
69  }
```

Presentation layer (function): add 3 chars (in this case `P-H`) with the string as head and send the new string to Session function.

```
55  void presentationLayer(string strl)
56  {
57      //char str[100]="P-H";
58      string str = "P-H"+strl;
59      //cout << str << "\n";
60      sessionLayer(str);
61  }
```

Session layer (function): add 3 chars (in this case `S-H`) with the string as head and send the new string to transport function.

```
48  void sessionLayer(string strl)
49  {
50      //char str[100]="S-H";
51      string str = "S-H"+strl;
52      //cout << str << "\n";
53      transportLayer(str);
54  }
```

Transport layer (function): add 3 chars (in this case `T-H`) with the string as head and send the new string to network function.

```
40  void transportLayer(string strl)
41  {
42      //char str[100]="T-H";
43      string str = "T-H"+strl;
44      //cout << str << "\n";
45      //printf("%d",strlen(str));
46      networkLayer(str);
47  }
```

Network layer (function): add 3 chars (in this case `N-H`) with the string as head and send the new string to data link function.

```
33  void networkLayer(string strl)
34  {
35      //char str[100]="N-H";
36      string str = "N-H"+strl;
37      //cout << str << "\n";
38      dataLinkLayer(str);
39  }
```

Datalink layer (function): add 3 chars (in this case `D-H`) with the string as head, more 3 chars (in this case `D-T`) with the string as Tail and send the new string to Physical function.

```
26  void dataLinkLayer(string strl)
27  {
28      //char str[100]="D-H",str2[]="D-T";
29      string str = "D-H"+strl+"D-T";
30      //cout << str << "\n";
31      physicalLayer(str);
32  }
```

Physical layer (function): add 4 chars (in this case `PH-H`) with the string as head and make the whole 150 chars to 8 bit binary and store this binary in 'temp.txt' file. The temp file will contain 125*8 = 1200 chars.

```
10  void physicalLayer(string strl)
11  {
12      //char str[100]="PH-H";
13      string str = "PH-H"+strl;
14      //cout << str << "\n";
15      //cout << str.length() << "\n\n";
16      //cout<<"\n"<<str.length();
17      ofstream temp;
18      temp.open("temp.txt",ios::app);
19
20      for(size_t i=0; i<str.size();i++)
21      {
22          temp << bitset<8>(str[i]);
23      }
24      temp.close();
25  }
```

receiver.cpp:

to create the OSI environment I create 7 more function which works as layers.

receiver (function): reads text file (temp.txt) and convert the binary string to char string and send the string to physical layer.

```
106  void receiver()
107  {
108      string txt;
109      ifstream temp("temp.txt");
110      while(getline(temp,txt))
111      {
112              //cout << str << '\n';
113      }
114      temp.close();
115      int txtlen = txt.size();
116      //cout << txtlen <<'\n';
117      int i=0;
118      //cout << str << '\n';
119      while(i!=(txtlen/1200)+1)
120      {
121          string str = txt.substr(1200*i,1200);
122          //cout << str << '\n';
123          setStringtoASCII(str);
124          i++;
125      }
126      //cout << "XX" << '\n';
127  }
 69  int binaryToDecimal(string n)
 70  {
 71      string num = n;
 72
 73      int dec_value = 0;
 74
 75      int base = 1;
 76
 77      int len = num.length();
 78      for (int i = len - 1; i >= 0; i--) {
 79
 80          if (num[i] == '1')
 81              dec_value += base;
 82          base = base * 2;
 83      }
 84
 85      return dec_value;
 86
 87  }
 89  void setStringtoASCII(string str)
 90  {
 91      int N = int(str.size());
 92
 93      string res = "";
 94
 95      for (int i = 0; i < N; i += 8) {
 96          int decimal_value
 97              = binaryToDecimal((str.substr(i, 8)));
 98
 99          res += char(decimal_value);
100      }
101
102      //cout << res << '\n';
103      RphysicalLayer(res);
104  }
```

Physical layer (function): already all the binary string is converted to a char string. Now in this layer 4 char from starting are removed and send to data link layer.

```
61  void RphysicalLayer(string strl)
62  {
63      int i= strl.length();
64      string str = strl.substr(4,i);
65      //cout << str<<'\n';
66      RdataLinkLayer(str);
67  }
```

Data link layer (function): remove the 1st and last total 6 chars and send to network layer.

```
52  void RdataLinkLayer(string strl)
53  {
54      int i= strl.length();
55      string str = strl.substr(3,i);
56      str = str.substr(0,str.size()-3);
57      //cout << str << '\n';
58      RnetworkLayer(str);
59  }
```

Network layer (function): remove the 1st 3 chars and send the string to transport layer.

```
44  void RnetworkLayer(string strl)
45  {
46      int i= strl.length();
47      string str = strl.substr(3,i);
48      //cout << str<<'\n';
49      RtransportLayer(str);
50  }
```

Transport layer (function): remove the 1st 3 chars and send the string to session layer.

```
36  void RtransportLayer(string strl)
37  {
38      int i= strl.length();
39      string str = strl.substr(3,i);
40      //cout << str<<'\n';
41      RsessionLayer(str);
42  }
```

Session layer (function): remove the 1st 3 chars and send the string to presentation layer.

```
28  void RsessionLayer(string strl)
29  {
30      int i= strl.length();
31      string str = strl.substr(3,i);
32      //cout << str<<'\n';
33      RpresentationLayer(str);
34  }
```

Presentation layer (function): remove the 1st 3 chars and send the string to application layer.

```
20  void RpresentationLayer(string strl)
21  {
22      int i= strl.length();
23      string str = strl.substr(3,i);
24      //cout << str<<'\n';
25      RapplicationLayer(str);
26  }
```

Application layer (function): remove the 1st 3 chars and print the output segment by segment and also store the full string in a text file (output.txt)

```
7   void RapplicationLayer(string strl)
8   {
9       int i= strl.length();
10      string str = strl.substr(3,i);
11      cout << "\nReceiver: ";
12      cout << str;
13      //RpresentationLayer(str);
14      ofstream output;
15      output.open("output.txt",ios::app);
16      output << str;
17      output.close();
18  }
```

Outputs:

If the text size is more than 125:

```
Sender  : East West University was established with interrelated objectives of augmenting national capacity for tertiary
 education to a larger number of students, and thereby contributing to the country's human capital development process.

Receiver: East West University was established with interrelated objectives of augmenting national capacity for tertiary
 education to a
Receiver:  larger number of students, and thereby contributing to the country's human capital development process.
```

temp.txt:

File Edit Format View Help


```
0101000001001000001011010100100001000100001011010100100001001110001011010100100001010100001011010100100001010011001011010100100001010000001011010
0110100101100001011100100111100100100000011001010110010001110101011000110110000101110100011010010110111011011100010000001110100011011110010000000
0110010001100101011101100110010010110110001101111011100000110110101100101011011100111010000100000011100000111001001101111011000110110010101110011
```


output.txt:

File Edit Format View Help

```
East West University was established with interrelated objectives of augmenting national capacity for tertiary education to a larger number of st
```

If the text size is less than 125:

```
Sender  : This is a text.

Receiver: This is a text.


Process returned 0 (0x0)   execution time : 7.550 s
Press any key to continue.
```

temp.txt:

File Edit Format View Help


```
0101000001001000001011010100100001000100001011010100100001001110001011010100100001010100001011010100100001010011001011010100100001010000001011010
```


output.txt:

File Edit Format View Help

```
This is a text.
```

Discursion:

While I was creating this application, I faced some problem such as I could not able to convert string to binary so I had to take help from online. I had to add extra 2 functions (`setStringtoASCII`, binaryToDecimal) to convert binary to string. Most of the problem I faced is now commented out. Mostly I faced those problems in “C” that’s why I started again in “C++”.