Green University of Bangladesh

Department of Computer Science and Engineering (CSE)

Faculty of Sciences and Engineering

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

Lab Report No: 01

Course Title: Data Communication Lab

Course Code: CSE 308   **Section:** PC-DD

Lab Experiment Name: Implementing character Stuffing and De-stuffing.

Student Details

Name	Id
Md. Romzan Alom	201902144

Lab Date: 15.06. 2022

Submission Date: 22.06.2022

Course Teacher's Name: Md. Nazmus Shakib

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

Title of the Lab Experiment: Implementing character Stuffing and De-stuffing.

Objectives / Aim:

We learn about Stuffing and De-stuffing from this experiment. We can take user input as a string and we can perform various operations on this input and show the result as output.

Introduction:

In byte stuffing, special byte that is basically known as ESC (Escape Character) that has predefined pattern is generally added to data section of the data stream or frame when there is message or character that has same pattern as that of flag byte. De- stuffing refers the controlled deletion of the stuffing characters from a stuffed digital signal in order to recover the original signal that existed prior to stuffing. *Note:* The deleted data may be transmitted via a separate low-traffic-capacity time slot to be compared with the stuffing data.

Problem:

Implement the Character stuffing and de-stuffing together where the system provides a choice to change the transmitted bit stream before de-stuffing. Consider the followings:

A B C D E F F E F D F E G

Where, F is the flag character and E is the escape character.

Problem analysis:

From the user string we need to find E and F. When we get E then our stuffing data will EE and when we get F then our stuffing data will EF otherwise we put same data. And the de-stuffing part, we will remove E where we get EE and also remove E where we get EF those are added in stuffing.

Like,

User input: ABCDERF

Stuffing: ABCDEEREF

De-stuffing: ABCDERF

Code:

```
#include<stdio.h>

int main(){

while(1){


int i=0,j=0;

char Input[100],Output_1[100],Romzan[100],Output_2[100];

printf("\nPlease, enter your data in UpperCase: ");

scanf("%s",Input);


printf("\n");
```

```c
while(Input[i]!='\0')
{
if(Input[i]=='F')
{
Output_1[j]='E';
Output_1[j+1]='F';
j+=2;
i++;
}
else if(Input[i]=='E')
{
Output_1[j]='E';
Output_1[j+1]='E';
j+=2;
i++;
}

else
{
Output_1[j++]=Input[i++];
```

```c
        }
    }
    Output_1[j]='\0';

    strcpy(Romzan,Output_1);
    i= 0;
    j= 0;

    while(Romzan[i]!='\0')
    {
    if(Romzan[i]=='E' && (Romzan[i+1]=='F'))
    {
    Output_2[j]='F';
    j++;
    i=i+2;
    }
    else if(Romzan[i]=='E' && Romzan[i+1]=='E')
    {
    Output_2[j]='E';
    j++;
    i=i+2;
```

```c
            }

        else

        {

        Output_2[j++]=Romzan[i++];

        }

    }

    Output_2[j]='\0';


    printf("After Stuffing: ");

    printf("%s",Output_1);

    printf("\n\n");


    printf("Do you want to see main data?");


    printf("\n1. Yes\t 2. No\n");

    int R;

    printf("Enter your opinion: ");

    scanf("%d",&R);
```

```c
switch (R){

case 1:

printf("\n\nAfter De-Stuffing: ");

printf("%s",Output_2);

break;


case 2:

break;


default:

printf("---Error---\n\n");

break;

}


printf("\n\n");

}

return 0;

}
```

Output:

```
"C:\Users\User\Documents\Data Communication lab report-1.exe"

Please, enter your data in UpperCase: ABCEDOFRER

After Stuffing: ABCEEDOEFREER

Do you want to see main data?
1. Yes    2. No

Enter your opinion: 1


After De-Stuffing: ABCEDOFRER


Please, enter your data in UpperCase: EROFFEOEEFFRG

After Stuffing: EEROEFEFEEOEEEEEFEFRG

Do you want to see main data?
1. Yes    2. No

Enter your opinion: 1


After De-Stuffing: EROFFEOEEFFRG


Please, enter your data in UpperCase: ABCODEFRO

After Stuffing: ABCODEEEFRO

Do you want to see main data?
1. Yes    2. No

Enter your opinion: 2


Please, enter your data in UpperCase:
```

To remove space error,

(use gets() and puts() function)

Code:


```c
#include<stdio.h>

int main(){
int i=0,j=0;
char Input[100],Output_1[100],Romzan[100],Output_2[100];

printf("\nPlease, enter your data in UpperCase: ");
gets(Input);

printf("\n");

while(Input[i]!='\0')
{
if(Input[i]=='F')
{
Output_1[j]='E';
```

```
Output_1[j+1]='F';


j+=2;

i++;

}

else if(Input[i]=='E')

{

Output_1[j]='E';

Output_1[j+1]='E';


j+=2;

i++;

}


else

{

Output_1[j++]=Input[i++];

}


}
```

```c
Output_1[j]='\0';

strcpy(Romzan,Output_1);

i= 0;

j= 0;

while(Romzan[i]!='\0')

{

if(Romzan[i]=='E' && (Romzan[i+1]=='F'))

{

Output_2[j]='F';

j++;

i=i+2;

}

else if(Romzan[i]=='E' && Romzan[i+1]=='E')

{

Output_2[j]='E';

j++;

i=i+2;

}
```

```c
else
{
Output_2[j++]=Romzan[i++];
}


}


Output_2[j]='\0';


printf("After Stuffing: ");

puts(Output_1);

printf("\n\n");


printf("Do you want to see main data?");


printf("\n1. Yes\t 2. No\n\n");

int R;

printf("Enter your opinion: ");

scanf("%d",&R);


switch (R){
```

```c
case 1:

printf("\n\nAfter De-Stuffing: ");

puts(Output_2);


break;


case 2:

break;


default:

printf("---Error---\n\n");

break;


}


printf("\n\n");


return 0;


}
```
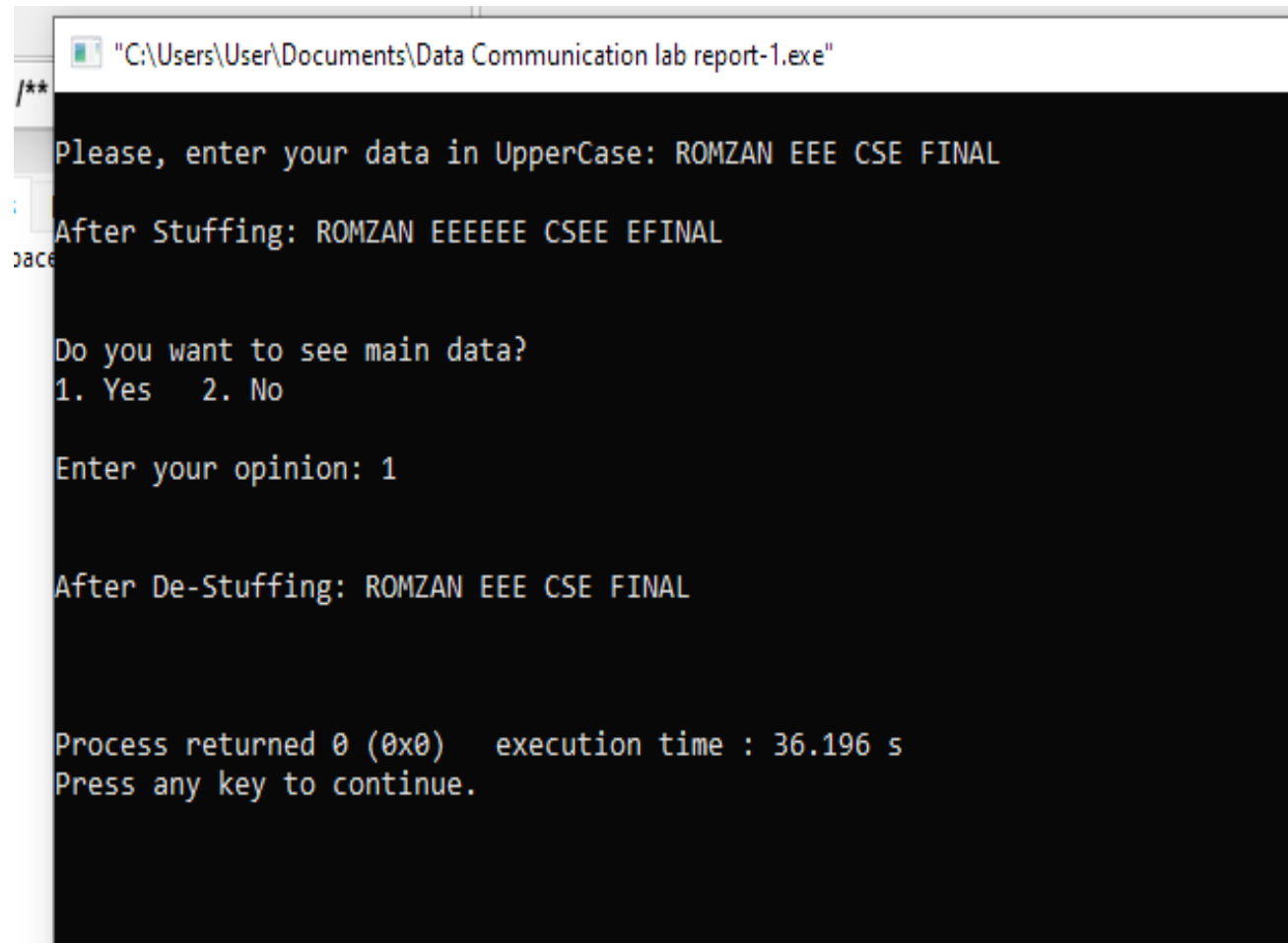
Output:

"C:\Users\User\Documents\Data Communication lab report-1.exe"

```
Please, enter your data in UpperCase: ROMZAN EEE CSE FINAL

After Stuffing: ROMZAN EEEEEE CSEE EFINAL


Do you want to see main data?
1. Yes    2. No

Enter your opinion: 1


After De-Stuffing: ROMZAN EEE CSE FINAL



Process returned 0 (0x0)    execution time : 36.196 s
Press any key to continue.
```

■■ "C:\Users\User\Documents\Data Communication lab report-1.exe"

```
Please, enter your data in UpperCase:

After Stuffing:


Do you want to see main data?
1. Yes   2. No

Enter your opinion: 1


After De-Stuffing:


Process returned 0 (0x0)   execution time : 5.739 s
Press any key to continue.
```

Data Communication lab report-1.c - Code::Blocks 20.03

File

<gl

Manag

◀ P

🏠 \1

■■ "C:\Users\User\Documents\Data Communication lab report-1.exe"

```
Please, enter your data in UpperCase: ROMZAN

After Stuffing: ROMZAN


Do you want to see main data?
1. Yes   2. No

Enter your opinion: 2


Process returned 0 (0x0)   execution time : 11.907 s
Press any key to continue.
```
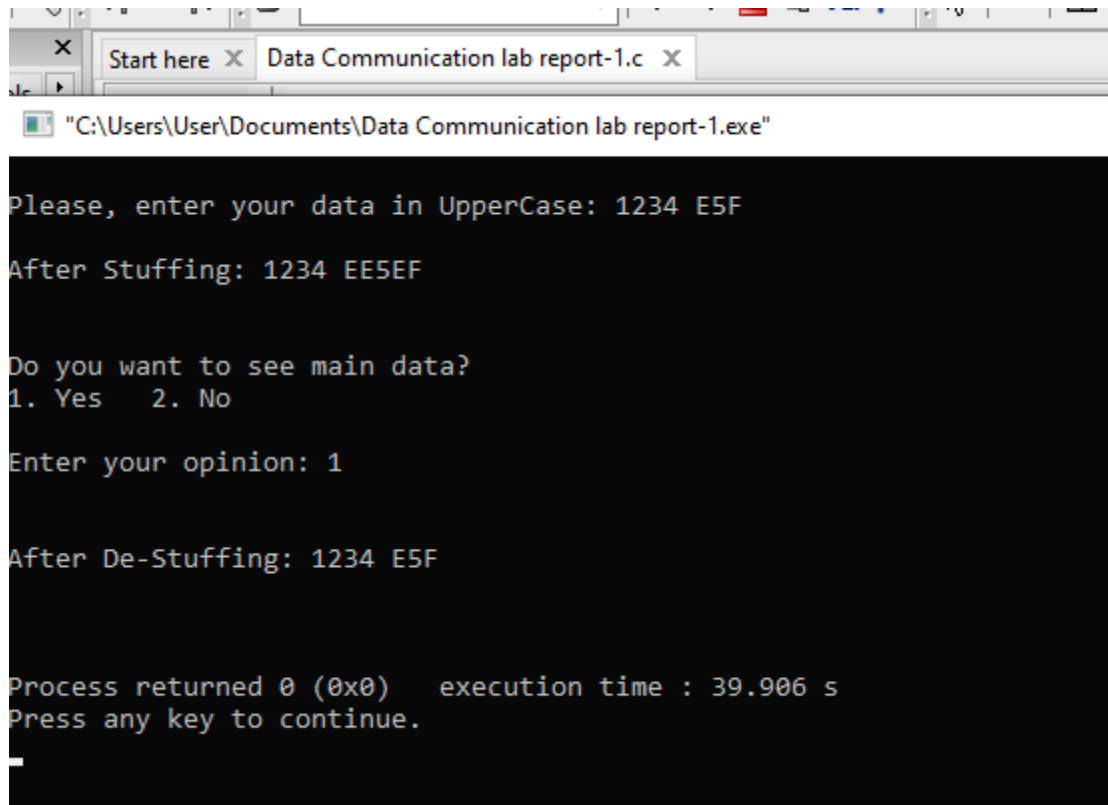
■ "C:\Users\User\Documents\Data Communication lab report-1.exe"

```
Please, enter your data in UpperCase: EEEEEEEEEE FFFFFFFFFF

After Stuffing: EEEEEEEEEEEEEEEEEEEE EFEFEFEFEFEFEFEFEFEF


Do you want to see main data?
1. Yes    2. No

Enter your opinion: 1


After De-Stuffing: EEEEEEEEEE FFFFFFFFFF



Process returned 0 (0x0)    execution time : 26.468 s
Press any key to continue.
```
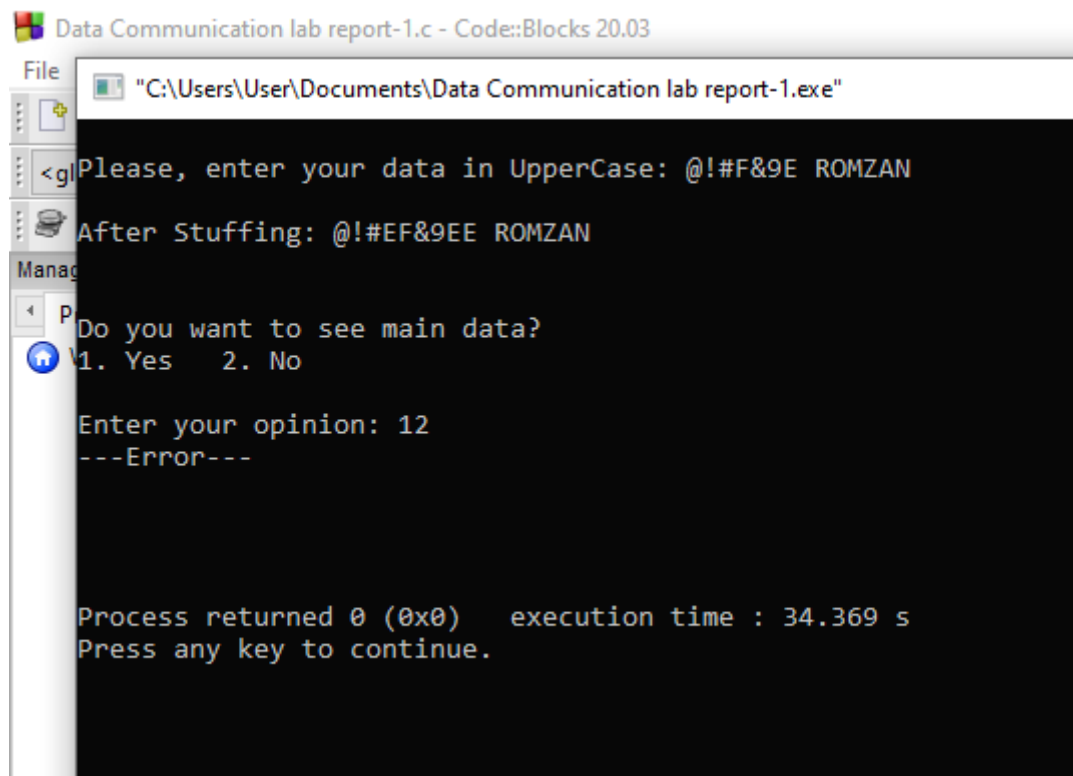
■ "C:\Users\User\Documents\Data Communication lab report-1.exe"

```
Please, enter your data in UpperCase: EFEFEF FEFEFE EEE FFF

After Stuffing: EEEFEEEFEEEF EFEEEFEEEFEE EEEEEE EFEFEF


Do you want to see main data?
1. Yes    2. No

Enter your opinion: 1


After De-Stuffing: EFEFEF FEFEFE EEE FFF



Process returned 0 (0x0)    execution time : 23.515 s
Press any key to continue.
```

"C:\Users\User\Documents\Data Communication lab report-1.exe"

```
Please, enter your data in UpperCase: 1234 E5F

After Stuffing: 1234 EE5EF


Do you want to see main data?
1. Yes    2. No

Enter your opinion: 1


After De-Stuffing: 1234 E5F



Process returned 0 (0x0)    execution time : 39.906 s
Press any key to continue.
```

Data Communication lab report-1.c - Code::Blocks 20.03

File

"C:\Users\User\Documents\Data Communication lab report-1.exe"

```
Please, enter your data in UpperCase: @!#F&9E ROMZAN

After Stuffing: @!#EF&9EE ROMZAN


Do you want to see main data?
1. Yes    2. No

Enter your opinion: 12
---Error---



Process returned 0 (0x0)    execution time : 34.369 s
Press any key to continue.
```

Analysis and Discussion:

1. From this lab, we knew stuffing and de-stuffing. And the working system of stuffing and de-stuffing. For stuffing, we add E in the input data. For de-stuffing, we deleted the extra bit and find the main input.

2. Due to temperature problem in our classroom, we can't do this lab with proper attention. So, we have some theoretical problem.

3. This lab is completely based on software. So it may have some Software and Mechanical errors.

4. When we use scanf() function to take user input then we can't access space and after the data of space. It is an error. That's why I use gets() function to access space and after the data of space.

5. From this problem, we use some array. When we operate that array index, we facing some problem.

6. Compile error.