==========

Name :**Lalita Sharma**

RNo :06

==========

Experiment :02

****

Aim:Write a program for implementing Bit Stuffing and character stuffing

Program:

```
#*************************************************************************
 # to convert string into binary
def toBinary(string):
    binary=""
    for char in string:
        ascii=ord(char)
        # print("ascii is {}".format(ascii))
        sum=0
        w=1
        while ascii != 0:
            d=ascii % 2
            sum=sum+d*w
            w=w*10
            ascii=ascii//2
        if len(str(sum))!=8:
            sum1='0'*(8-len(str(sum))) +str(sum)
        binary=binary+str(sum1)
    return binary
#*************************************************************************
# bit stuffing
def DataAfterBitStuffing(b_str):
    stuffed=""
    count=0
```

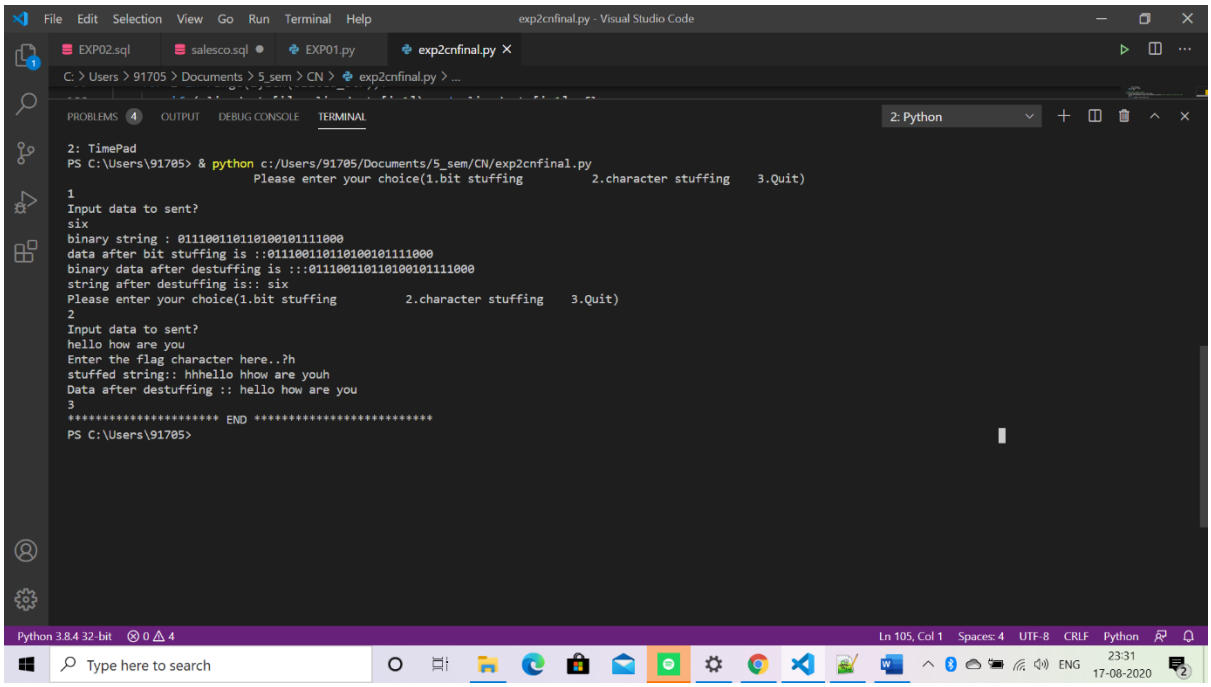
```
    indx=-1
    for char in b_str:
        indx+=1
        if int(char)==1:
            count=count+1
            stuffed+=char
        elif int(char)!=1:
            count=0
            stuffed+=char

        if count==5:
            print("index is {}".format(indx))
            stuffed=stuffed[:indx+1] +'0'   #adding a 0
            indx+=1
            count=0

    return stuffed
#*****************************************************************************
# returns a destuffed binary string
def destuffing(stuffed_str):
    count=0
    destuff=''
    highlight=1  #to skip a character after 5 1's
    for char in stuffed_str:
        if highlight==60:
            highlight=1
            continue
        if int(char)==1:
            count+=1
            destuff+=char
        elif int(char)!=1:
            count=0
            destuff+=char
        if count==5:
```

```
            count=0
            highlight=60
    return destuff

#*****************************************************************************
# to convert destuffed binary string to actual string
def Back_to_str(binary_str):
    len1=8
    ori=''
    ini=0
    range1=len(binary_str)//8
    for i in range(range1):
        sum=0
        w=1
        one_char=binary_str[ini:len1]
        one_char=int(one_char)
        while int(one_char)!=0:
            d=one_char%10
            sum=sum+d*w
            w=w*2
            one_char=one_char//10
        ori+=chr(sum)
        ini=len1
        len1+=8
    return ori
#*****************************************************************************#
returns a character stuffed string
def stuffed_str_characterstuffing(string ,flag):
    ret=''
    for i in range(len(string)):
        if string[i]==flag :
            ret+=flag
        ret+=string[i]
    ret= flag+ret +flag
```

```
    return ret

#*****************************************************************************

# returns the original string

def destuffing_char(stuffed_str_char,flag):

    sliced_str=stuffed_str_char[1:len(stuffed_str_char)-1]  #to remove first and
last char

    ret=''

    for i in range(1,len(sliced_str)):

        if (sliced_str[i]==sliced_str[i-1]) and sliced_str[i-1]==flag:

            continue

        ret=ret+sliced_str[i-1]

    return ret+sliced_str[-1]


#=============================================================================




#        ********** driver function ********************

while(1):
    choice=int(input("Please enter your choice(1.bit stuffing \t 2.character
stuffing\t 3.Quit)\n"))
    if choice ==1:
        string=input("Input data to sent?\n")
        binary_str=toBinary(string)
        print("binary string : {}".format(binary_str))
        stuffed_str=DataAfterBitStuffing(binary_str)
        print("data after bit stuffing is ::{}".format(stuffed_str))
        binary_str2=destuffing(stuffed_str)
        print("binary data after destuffing is :::{}".format(binary_str2))
        originalstr=Back_to_str(binary_str2)
        print("string after destuffing is:: {}".format(originalstr))

    elif choice==2:
        string=input("Input data to sent?\n")
        flag=input("Enter the flag character here..?")
        stuffed_str_char=stuffed_str_characterstuffing(string,flag)
        print("stuffed string::",stuffed_str_char)
        final_destuff_char=destuffing_char(stuffed_str_char,flag)
        print("Data after destuffing ::",final_destuff_char)
    else:
         print("********************** END ************************")
         break
```

OUTPUT

==========


File Edit Selection View Go Run Terminal Help
exp2cnfinal.py - Visual Studio Code
EXP02.sql
salesco.sql
EXP01.py
exp2cnfinal.py
C: > Users > 91705 > Documents > 5_sem > CN > exp2cnfinal.py > ...
PROBLEMS 4
OUTPUT
DEBUG CONSOLE
TERMINAL
2: Python
2: TimePad
PS C:\Users\91705> & python c:/Users/91705/Documents/5_sem/CN/exp2cnfinal.py
Please enter your choice(1.bit stuffing 2.character stuffing 3.Quit)
1
Input data to sent?
six
binary string : 011100110110100101111000
data after bit stuffing is ::011100110110100101111000
binary data after destuffing is :::011100110110100101111000
string after destuffing is:: six
Please enter your choice(1.bit stuffing 2.character stuffing 3.Quit)
2
Input data to sent?
hello how are you
Enter the flag character here..?h
stuffed string:: hhhello hhow are youh
Data after destuffing :: hello how are you
3
******************** END ************************
PS C:\Users\91705>
Python 3.8.4 32-bit
Ln 105, Col 1
Spaces: 4
UTF-8
CRLF
Python
Type here to search
ENG
23:31
17-08-2020