	Exp. No: 1 **Date**

Aim: Write a C program that contains a string(charpointer) with a value \HelloWorld'.The program should XOR each character in this string with 0 and display the result

Program:

```
#include<stdio.h>
#include<string.h>
void main(){
char inputString[100]=”Hello World”;
char outputString[100];
int i;
int stringLength=strlen(inputString);
for(i=0;i<=stringLength;i++)
{
outputString[i] = inputString[i]^0;
printf(“%c”,outputString[i]);
}
}
```

Output:

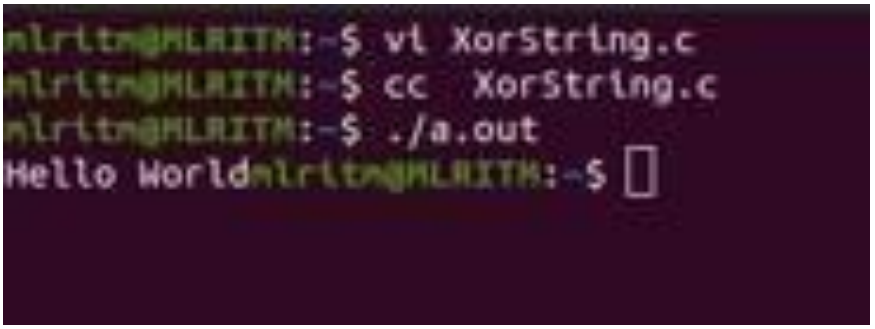
	Exp. No: 2 Date

Aim: Write a C program that contains a string(charpointer)with a value 'HelloWorld'. The program should AND or and XOR each character in this string with 127 and display the result.

Program:

```
#include<stdio.h>
#include<string.h>
void main(){
char inputString[100]="Hello World";
char outputString[100];
int j;
int stringLength=strlen(inputString);
for(j=0;j<=stringLength;j++)
{
outputString[i] = inputString[i]^127;
printf("%c",outputString[j]);
}
}
```

Output:



	Exp. No : 3 **Date:**

Aim: Write a Java program to perform encryption and decryption using the following algorithms:

Program:

a) CeaserCipher

```
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
import java.util.Scanner;
 class CeaserCipher {
static Scanner sc=new Scanner(System.in);
static BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
public static void main(String[] args) throws IOException {
//TODO code application logic here
System.out.print("Enter any String: ");
String str = br.readLine();
System.out.print(" \n Enter the Key:"); int key = sc.nextInt();
String encrypted = encrypt(str, key);
System.out.println(" \n Encrypted String is: " +encrypted);
String decrypted = decrypt(encrypted, key);
System.out.println(" \n Decrypted String is: "+decrypted);
System.out.println(" \n");
}
public static String encrypt(String str, int key)
{ String encrypted = "";
for(int i = 0; i < str.length();
i++) { int c = str.charAt(i);
if (Character.isUpperCase(c)) {
c = c + (key % 26);
if (c > 'Z')
c = c - 26;}
else if (Character.isLowerCase(c)){
```

	Exp. No : Date:

```
c=c+(key % 26);
if (c >'z')

c = c - 26;}
encrypted += (char)c;
}
return encrypted;
}
public static String decrypt(String str, int key)
{ String decrypted = "";
for(int i = 0; i < str.length();
i++) { int c = str.charAt(i);
if (Character.isUpperCase(c)) {
c = c - (key % 26);
if (c < 'A')
c = c+ 26;
else if (Character.isLowerCase(c)){
c = c - (key % 26);
if (c < 'a')
c = c+ 26;
decrypted += (char)c;
}
}
}
return decrypted;
}
}
```

Output:

	Exp. No: 3b **Date**

Program:
SubstitutionCipher

```
import java.io.*;
importjava.util.*;
public class SubstitutionCipher{
static Scanner sc = new Scanner(System.in);
static BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
public
static void main(String[] args) throws IOException {
// TODO code application logic her e
Stringa="abcdefghijklmnopqrstuvwxyz";
Stringb= "zyxwvutsrqponmlkjihgfedcba";
System.out.print("Enter any string: ");
String str = br.readLine();
Stringdecrypt="";
char c;
for(int i=0;i<str.length();i++)
{
c =
str.charAt(i);intj=
a.indexOf(c);
decrypt = decrypt+b.charAt(j);
}
System.out.println("The encrypted data is: " +decrypt);
}
}
```

	Exp. No:3b **Date**

Output:

```
java -cp /tmp/f7egljfPAW SubstitutionCipher
Enter any string   aceho
aceho
The encrypted data is: zxvil
```

Exp.No:3c
Date

Program:

a) HillCipher

```
import java.io.*;
import java.util.*;
import java.io.*;
publicclass HillCipher{
static float[][] decrypt = new float[3][1];
static float[][] a = new float[3][3];
static float[][] b = new float[3][3];
static float[][] mes = new float[3][1];
static float[][] res = new float[3][1];
static BufferedReader br= newBufferedReader(new
InputStreamReader(System.in));
static Scanner sc = new Scanner(System.in);
public static void main(String[] args) throws IOException{
getkeymes();
for(int i=0;i<3;i++)
for(int j=0;j<1;j++)
for(int k=0;k<3;k++) {
res[i][j]=res[i][j]+a[i][k]*mes[k][j]; }
System.out.print(" \nEncrypted string is : ");
for(int i=0;i<3;i++) {
System.out.print((char)(res[i][0]%26+97));
res[i][0]=res[i][0];
}
inverse();
for(int i=0;i<3;i++)
for(int j=0;j<1;j++)
for(int k=0;k<3;k++) {
decrypt[i][j] = decrypt[i][j]+b[i][k]*res[k][j]; }
System.out.print(" \nDecrypted string is : ");
```

Exp.No:3c
Date

```
for(int i=0;i<3;i++){ System.out.print((char)(decrypt[i][0]%26+97));
}
System.out.print(" \n");
}
public static void getkeymes() throws IOException{
System.out.print(" \nEnter a 3 letter string: ");
String msg = br.readLine();
for(int i=0;i<3;i++)
mes[i][0] = msg.charAt(i)-97;
}
public static void inverse() {
float p,q;
float[][] c = a;
for(int i=0;i<3;i++)
for(int j=0;j<3;j++) {
a[i][j]=sc.nextFloat();
if(i==j)
b[i][j]=1;
else b[i][j]=0;
}
for(int k=0;k<3;k++) {
for(int i=0;i<3;i++) {
p = c[i][k];
q = c[k][k]; for(int j=0;j<3;j++)
{
if(i!=k)
{
c[i][j] = c[i][j]*q -p*c[k][j];
b[i][j] = b[i][j]*q-p*b[k][j];
}
}
```

Exp.No:3c
Date

```
}
}
for(int i=0;i<3;i++)
for(int j=0;j<3;j++)
{
b[i][j] = b[i][j]/c[i][i];
}
System.out.println("");
System.out.println(" \nInverse Matrix is :");
for(int i=0;i<3;i++) {
for(int j=0;j<3;j++)
System.out.print(b[i][j] + "");
System.out.print(" \n");
}
}
}
```

Output:

```
$ java -cp /tmp/cq7PDDaDiW HillCipher
Enter a 3 letter string: hai
Encrypted string is :fdx
Inverse Matrix is :
0.083333336 0.41666666 -0.33333334
-0.41666666 -0.083333336 0.6666667
0.5833333 -0.083333336 -0.33333334
Decrypted string is: hai
```

	Exp. No: 4 Date

Aim: Write a Java program to implement the DES algorithm logic

Program:

```
import java.util.*;
import java.io.BufferedReader;
import java.io.InputStreamReader;
import java.security.spec.KeySpec;
import javax.crypto.Cipher;
import javax.crypto.SecretKey;
import javax.crypto.SecretKeyFactory;
import javax.crypto.spec.DESedeKeySpec;
import sun.misc.BASE64Decoder;
import sun.misc.BASE64Encoder;
public class DES {
private static final String UNICODE_FORMAT = "UTF8";
public static final String DESEDE_ENCRYPTION_SCHEME = "DESede";
private KeySpec myKeySpec;
private SecretKeyFactory mySecretKeyFactory;
private Cipher cipher;
byte[] keyAsBytes;
private String myEncryptionKey;
private String myEncryptionScheme;
SecretKey key;
staticBufferedReaderbr= newBufferedReader(new
InputStreamReader(System.in));
public DES() throws Exception {
myEncryptionKey= "ThisIsSecretEncryptionKey"; myEncryptionScheme =
DESEDE_ENCRYPTION_SCHEME;
keyAsBytes =myEncryptionKey.getBytes(UNICODE_FORMAT);
myKeySpec= new DESedeKeySpec(keyAsBytes);
mySecretKeyFactory = SecretKeyFactory.getInstance(myEncryptionScheme);
cipher =Cipher.getInstance(myEncryptionScheme);
```

	Exp. No: 4 Date

```
key =mySecretKeyFactory.generateSecret(myKeySpec);
}

public String encrypt(StringunencryptedString)
{
String encryptedString = null;
try {
cipher.init(Cipher.ENCRYPT_MODE, key);
byte[] plainText = unencryptedString.getBytes(UNICODE_FORMAT); byte[]
encryptedText = cipher.doFinal(plainText);
BASE64Encoder base64encoder = new BASE64Encoder();
encryptedString = base64encoder.encode(encryptedText);
}
catch (Exception e) {
e.printStackTrace(); }
return encryptedString;
}
public String decrypt(StringencryptedString)
{
String decryptedText=null;
try {
cipher.init(Cipher.DECRYPT_MODE, key);
BASE64Decoderbase64decoder=newBASE64Decoder();
byte[] encryptedText = base64decoder.decodeBuffer(encryptedString);
byte[] plainText = cipher.doFinal(encryptedText);
decryptedText= bytes2String(plainText);
}
catch (Exception e) {
e.printStackTrace(); }
return decryptedText; }
private static String bytes2String(byte[] bytes)
{ StringBufferstringBuffer =new
```

	Exp. No: 4 Date

```
StringBuffer();
for (int i = 0; i <bytes.length;i++)
{
stringBuffer.append((char) bytes[i]);
}

returnstringBuffer.toString();
}
public static void main(String args []) throws Exception
{
System.out.print("Enter the string: ");
DES myEncryptor= new DES();
String stringToEncrypt =br.readLine();
Stringencrypted= myEncryptor.encrypt(stringToEncrypt); Stringdecrypted=
myEncryptor.decrypt(encrypted); System.out.println(" \nString To Encrypt: "
+stringToEncrypt); System.out.println("\nEncrypted Value :"+encrypted);
System.out.println(" \nDecrypted Value : " +decrypted);
System.out.println("");
}
}
```

Output:

```
javac /tmp/WoSptsLv8X/DES.java
Enter the string: Welcome String
To Encrypt: Welcome
Encrypted Value : BPQMwcOwKvg=
Decrypted Value : Welcome
```

	Exp. No: 5 Date

Aim: Write a C/JAVA program to implement the BlowFish algorithm logic.

Program:

```
import java.io.*;
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.security.Key;
import javax.crypto.Cipher;
import javax.crypto.CipherOutputStream;
import javax.crypto.KeyGenerator;
import sun.misc.BASE64Encoder;
public class BlowFish{
public static void main(String[] args) throws Exception {
KeyGeneratorkeyGenerator =
KeyGenerator.getInstance("Blowfish");
keyGenerator.init(128);
Key secretKey = keyGenerator.generateKey();
Cipher cipherOut = Cipher.getInstance("Blowfish/ CFB/ NoPadding");
cipherOut.init(Cipher.ENCRYPT_MODE, secretKey); BASE64Encoder encoder = new
BASE64Encoder();
byte iv[] = cipherOut.getIV();
if (iv != null) {
System.out.println("Initialization Vector of the Cipher: " + encoder.encode(iv));
}
FileInputStream fin = new FileInputStream("inputFile.txt");
FileOutputStreamfout = new FileOutputStream("outputFile.txt");
CipherOutputStreamcout = new CipherOutputStream(fout,cipherOut);
int input = 0;
while ((input = fin.read()) != -1)
{
cout.write (input);
}
```

	Exp. No: 5 Date

```
fin.close(); cout.close();
}
}
```

Output:

```
$ javac /tmp/WoSptsLv8X/BlowFish.java
Initialization Vector of the Cipher:
dI1MXzW97oQ= Contents of inputFile.txt: Hello
World
Contents of outputFile.txt: ùJÖ˜ NåI "
```