import java.util.\*;

class msg

{

int a=97;

char all[]=new char[27];

msg()

{

for(int i=0;i < 26;i++)

{

all[i]=(char)a;

a++;

}

}

int Ipos(char c)

{

int i=0;

for(;i < 26;i++)

{

if(all[i]==c)

{

break;

}

}

return i;

}

char Cpos(int c)

{

int i=0;

for(;i < c;i++)

{

}

return all[i];

}

}

class OneTimePadCipherImplementation

{

String Encryption(String plaintext,String key)

{

plaintext=plaintext.toLowerCase();

msg m1=new msg();

int pt[]=new int[plaintext.length()];

int k[]=new int[key.length()];

int ct[]=new int[plaintext.length()];

for(int i=0;i < plaintext.length();i++)

{

pt[i]=m1.Ipos(plaintext.charAt(i));

}

for(int i=0;i < key.length();i++)

{

k[i]=m1.Ipos(key.charAt(i));

}

int j=0;

for(int i=0;i < plaintext.length();i++)

{

ct[i]=pt[i]+k[j];

j++;

if(j==key.length())

j=0;

if(ct[i]>26)

ct[i]=ct[i]%26;

}

String cipher="";

for(int i=0;i < plaintext.length();i++)

{

cipher+=m1.Cpos(ct[i]);

}

return cipher;

}

String Decryption(String ciphertext,String key)

{

String plaintext="";

msg m1=new msg();

int pt[]=new int[ciphertext.length()];

int k[]=new int[key.length()];

int ct[]=new int[ciphertext.length()];

for(int i=0;i < ciphertext.length();i++)

{

ct[i]=m1.Ipos(ciphertext.charAt(i));

}

for(int i=0;i < key.length();i++)

{

k[i]=m1.Ipos(key.charAt(i));

}

int j=0;

for(int i=0;i < ciphertext.length();i++)

{

pt[i]=ct[i]-k[j];

j++;

if(j==key.length())

j=0;

if(pt[i] < 0)

pt[i]+=26;

}

String cipher="";

for(int i=0;i < ciphertext.length();i++)

{

plaintext+=m1.Cpos(pt[i]);

}

return plaintext;

}

}

class OneTimePad

{

public static void main(String args[])throws Exception

{

String plaintext,key;

Scanner scn=new Scanner(System.in);

System.out.println("Enter plaintext:");

plaintext=scn.nextLine();

System.out.println("Enter key:");

key=scn.nextLine();

OneTimePadCipherImplementation OneTimePad=new OneTimePadCipherImplementation();

String ciphertext=OneTimePad.Encryption(plaintext,key);

System.out.println("Encrpted text is:"+ciphertext);

System.out.println("Decrypted text is:"+OneTimePad.Decryption(ciphertext,key));

}

}

Output:

