

PRACTICAL 1

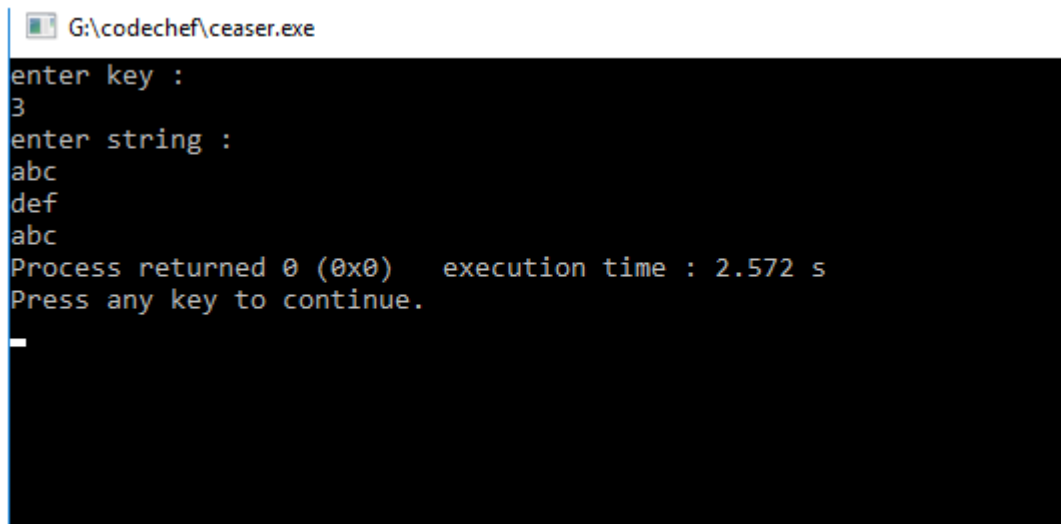
AIM: To implement Caesar cipher encryption-decryption.

CODE:

```
#include<iostream>
using namespace std;
void enc(string s,int key)
{
    int n=s.length();
    string a="abcdefghijklmnopqrstuvwxyz";
    for(int i=0;i<n;i++)
    {
        for(int j=0;j<26;j++)
        {
            if(s[i]==a[j])
            {
                s[i]=a[(j+key)%26];
                break;
            }
        }
    }
    cout<<s<<endl;
}
void dec(string s,int key)
{
    int n=s.length();
    string a="abcdefghijklmnopqrstuvwxyz";
    // string a="zyxwvutsrqponmlkjihgfedcba";
    for(int i=0;i<n;i++)
    {
        for(int j=0;j<26;j++)
        {
            if(s[i]==a[j])
            {
                s[i]=a[(j-key)%26];
                break;
            }
        }
    }
    cout<<"";
    s="abc";
    cout<<s;
}
int main()
```

```
{  
    int key;  
    cout<<"enter key : "<<endl;  
    cin>>key;  
    string s;  
    cout<<"enter string : "<<endl;  
    cin>>s;  
    enc(s,key);  
    //cout<<s;  
    dec(s,key);  
    return 0;  
}  
}
```

Output:



The screenshot shows a Windows command prompt window with the title bar "G:\codechef\ceaser.exe". The program prompts the user to "enter key :", and the user enters "3". It then prompts "enter string :", and the user enters "abc". The program outputs "def" and "abc". Finally, it displays "Process returned 0 (0x0) execution time : 2.572 s" and "Press any key to continue." before waiting for a key press.

```
G:\codechef\ceaser.exe  
enter key :  
3  
enter string :  
abc  
def  
abc  
Process returned 0 (0x0) execution time : 2.572 s  
Press any key to continue.  
_
```

PRACTICAL 2

AIM: To implement Monoalphabetic cipher encryption-decryption.

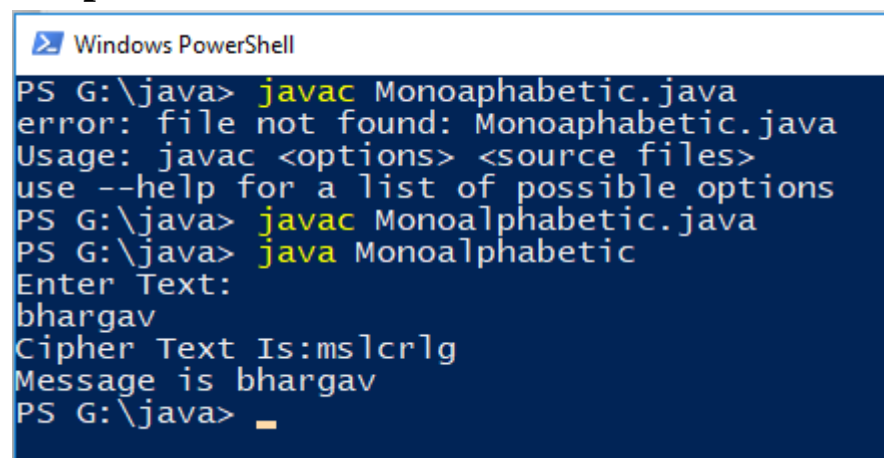
CODE:

```
import java.util.Scanner;

class Monoalphabetic
{
    public static void main(String args[])
    {
        String cipher="";
        Scanner sc=new Scanner(System.in);
        String str="abcdefghijklmnopqrstuvwxyz";
        String k="lmnopqrstuvwxyzabcdefghijklmnopqrstuvwxyz";
        System.out.println("Enter Text:");
        String message =sc.next();
        int len=message.length();
        for(int i=0;i<len;i++)
        {
            int p=0;
            for(int j=0;j<26;j++)
            {
                if(message.charAt(i)== str.charAt(j))
                {
                    p=j;
                    break;
                }
            }
            cipher=cipher+k.charAt(p);
        }
        System.out.println("Cipher Text Is:"+cipher);
        message ="";
        len=cipher.length();
        for(int i=0;i<len;i++)
        {
```

```
        int p=0;
        for(int j=0;j<26;j++)
        {
            if(cipher.charAt(i)== k.charAt(j))
            {
                p=j;
                break;
            }
        }
        message=message+str.charAt(p);
    }
    System.out.println("Message is "+ message);
}
```

Output:



```
Windows PowerShell
PS G:\java> javac Monoalphabetic.java
error: file not found: Monoalphabetic.java
Usage: javac <options> <source files>
use --help for a list of possible options
PS G:\java> javac Monoalphabetic.java
PS G:\java> java Monoalphabetic
Enter Text:
bhargav
Cipher Text Is:mslcrlg
Message is bhargav
PS G:\java> _
```

PRACTICAL 3

AIM: To implement Playfair cipher encryption-decryption.

CODE:

```
import java.util.Scanner;

public class Playfr1
{
    public static void main(String args[])
    {
        Scanner s=new Scanner(System.in);

        System.out.println("enter key");

        String kyd=s.nextLine();

        System.out.println("enter plan text");

        String plt=s.nextLine();

        String finl="";

        String alb="abcdefghijklmnopqrstuvwxyz";

        //    final char pt[] =
        {'a','b','c','d','e','f','g','h','i','j','k','l','m','n','o','p','q','r','s','t','u','v','w','x','y','z'};

        String sf=kyd.concat(alb);

        //System.out.println(sf);

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

            if(!finl.contains(String.valueOf(sf.charAt(i))))

            {

                if('j'!=sf.charAt(i))

                {

                    finl+=String.valueOf(sf.charAt(i));

                }

            }

        }

    }
}
```

```
        }

    }

    //System.out.println(finl);

    char k[]=new char[26];

    char mt[][]=new char[5][5];

    int c=0;

    for(int i=0;i<5;i++)

    {

        for(int j=0;j<5;j++)

        {

            mt[i][j]=finl.charAt(c);

            c++;

        }

    }

    for(int i=0;i<5;i++)

    {

        for(int j=0;j<5;j++)

        {

            System.out.print(mt[i][j]+"\\t");

        }

        System.out.println();

    }

    int rrr;

    String bln="";

    System.out.println(bln);

    String ss="";
```

```
for(int i=0;i<plt.length();i=i+2)
{
    ss+=plt.charAt(i);
    if(i==plt.length()-1)
    {
        ss+=plt.charAt(i);
        break;
    }
    if(plt.charAt(i)==plt.charAt(i+1))
    {
        ss+='x';
    }
    ss+=plt.charAt(i+1);
}
if(ss.length()%2!=0)
    ss+='x';
System.out.println(ss);

int r1=0;
int r2=0;
int c1=0;
int c2=0;
int x=ss.length();
char[] cc=new char[x];
for(int l=0;l<ss.length();l=l+2)
{
    for(int i=0;i<5;i++)
    {
        for(int j=0;j<5;j++)
```

```
{  
  
    if(ss.charAt(l)==mt[i][j])  
    {  
  
        r1=i;  
        c1=j;  
  
        //System.out.println(r1);  
        //System.out.println(c1);  
  
    }  
    if(ss.charAt(l+1)==mt[i][j])  
    {  
  
        r2=i;  
        c2=j;  
  
        //System.out.println(r2);  
        //System.out.println(c2);  
  
    }  
    if(r1==r2)  
    {  
  
        cc[l]=mt[r1][(c1+1)%4];  
        cc[l+1]=mt[r2][(c2+1)%4];  
  
    }  
    if(c1==c2)  
    {  
  
        cc[l]=mt[(r1+1)%4][c1];  
        cc[l+1]=mt[(r2+1)%4][c2];  
  
    }  
    if(r1!=r2 && c1!=c2)  
    {  
  
        cc[l]=mt[r1][c2];
```



```

        cc[l+1]=mt[r2][c1];

    }

}

}

}

for(int i=0;i<ss.length();i++)
{
    System.out.println(cc[i]);
}

}

}

```

```

Windows PowerShell
PS C:\Users\Admin\Downloads\IS> javac Playfr1.java
PS C:\Users\Admin\Downloads\IS> java Playfr1
enter key
bvm
enter plan text
aarraarr
b      v      m      a      c
d      e      f      g      h
i      k      l      n      o
p      q      r      s      t
u      w      x      y      z
axarxraxarxr
bvm
aarraarr

```