**PRACTICAL-2**

**AIM: Implement monoalphabetic cipher encryption and decryption algorithm.**

**EXPLANATION:**

* Monoalphabetic cipher is a substitution cipher in which for a given key, the cipher alphabet for each plain alphabet is fixed throughout the encryption process.
* For example, if ‘A’ is encrypted as ‘D’ , for any number of occurence in that plain text, ‘A’ will always get encrypted to ‘D’.
* There are many different monoalphabetic substitution ciphers, in fact infinitely many, as each letter can be encrypted to any symbol, not just another letter.

**CODE**:

#include<stdio.h>

#include<string.h>

#include<conio.h>

char pt[30],c[27],ct[30];

int i,j,index;

void encrypt(char ct[],char c[]);

void decrypt(char pt[],char c[]);

int main(){

printf("enter your plaintext:");

gets(pt);

printf("enter your key:");

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

{

printf("%c-",i+97);

c[i]=getch();

printf("%c , ",c[i]);

}

for(i=0;i<strlen(pt);i++)

{

index=pt[i]-97;

ct[i]=c[index];

}

encrypt(ct,c);

decrypt(pt,c);

return 0;

}

void encrypt(char ct[], char c[]){

printf("\n\ncipher Text is : ");

for(i=0;i<strlen(pt);i++)

{ printf("%c",ct[i]);}

for(i=0;i<strlen(pt);i++)

{ ct[i]=pt[i];}

}

void decrypt(char pt[], char c[]) {

printf("\n\nPlain Text is : ");

for(i=0;i<strlen(pt);i++)

{ printf("%c",ct[i]);}

}

**OUTPUT**:



