CSI 3002 - Applied Cryptography and Network Security Lab
Digital Assessment -I

22MIC0130
A Vijay Pavan

1. Write a C/C++/JAVA program to perform encryption and decryption using the following algorithms.

- Ceasar cipher
- Playfair cipher (Encryption only)
- Vignere cipher
- Hill cipher

Note:

1. Ceasar cipher plain text is: drapjabdulkalam key: 4
2. Playfair cipher plain text is: velloreinstituteoftechnology key: Student
3. Vignere cipher plain text is: Computer Science and Engineering, Key: Subject
4. Hill cipher performs (3x3) matrix with plain text is: ACTDOG

$$\text{Key} = \begin{vmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 11 & 9 & 8 \end{vmatrix}$$

- Ceasar cipher

Code :

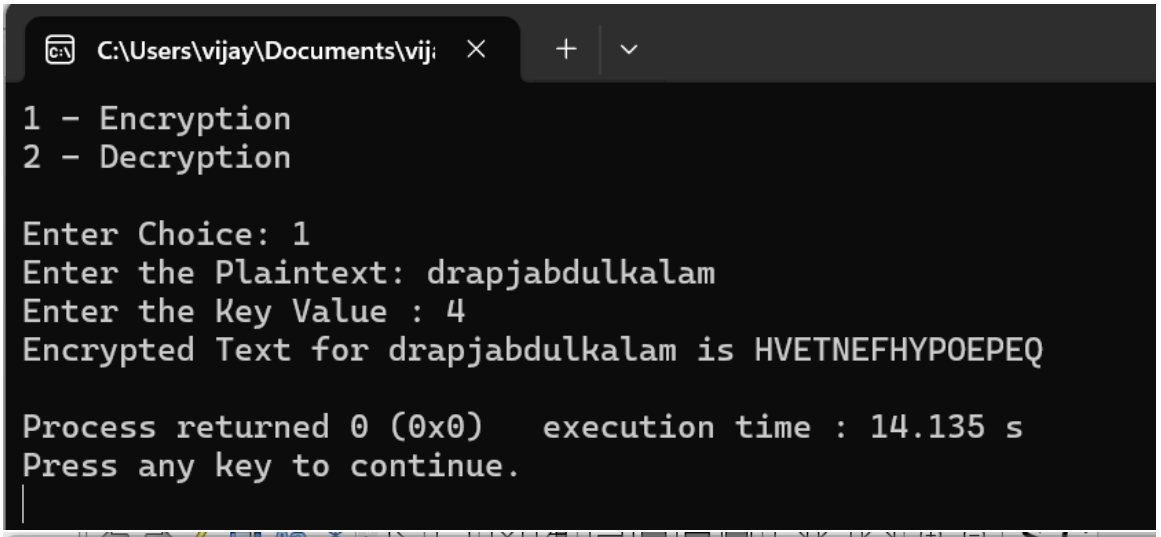
Start here | vijaycrypto.c

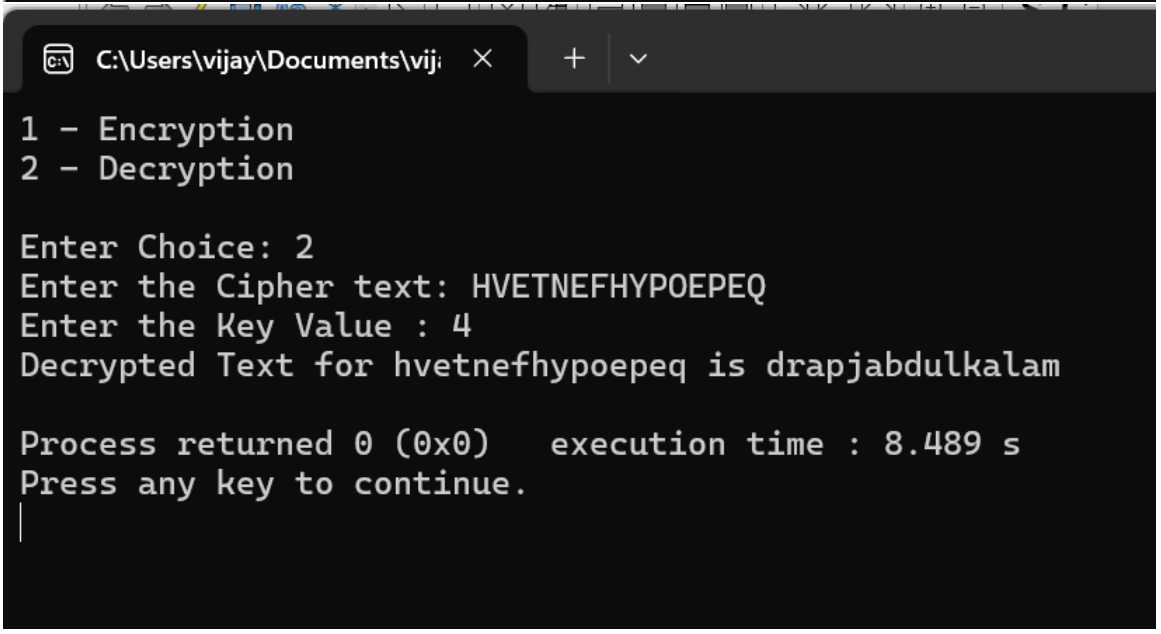
```c
#include <stdio.h>
#include <string.h>
int main()
{
    char p[100], c[100];
    int key, i, choice;
    printf("1 - Encryption\n2 - Decryption\n\nEnter Choice: ");
    scanf("%d", &choice);
    if (choice == 1)
    {
        printf("Enter the Plaintext: ");
        scanf("%s", p);
    for(i = 0; i<strlen(p); i++)
    {
        if(p[i]>='A' && p[i]<='Z')
        {
            p[i] = p[i] + 32;
        }
    }
    printf("Enter the Key Value : ");
    scanf("%d", &key);
    for (i = 0; i<strlen(p); i++)
    {
        if (p[i] >= 'a' && p[i] <= 'z')
        {
            c[i] = ((p[i] + key));
            if(c[i]>'z')
            {
                c[i]-=26;
            }
        }
        c[i]-=32;
    }
    c[i]='\0';
```

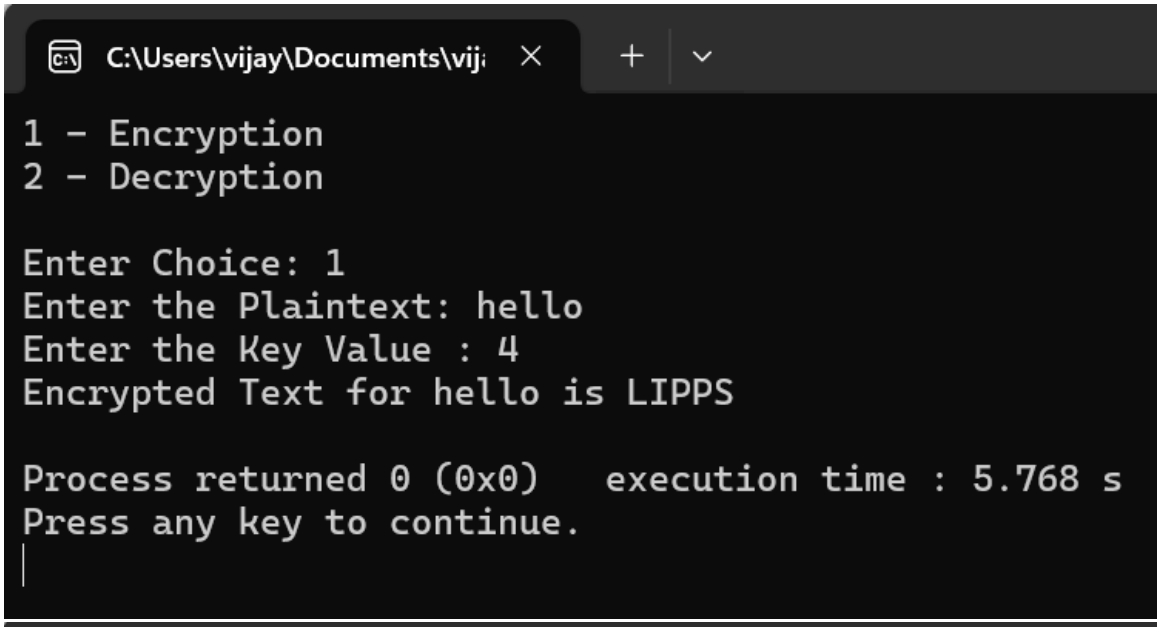
Start here | *vijaycrypto.c

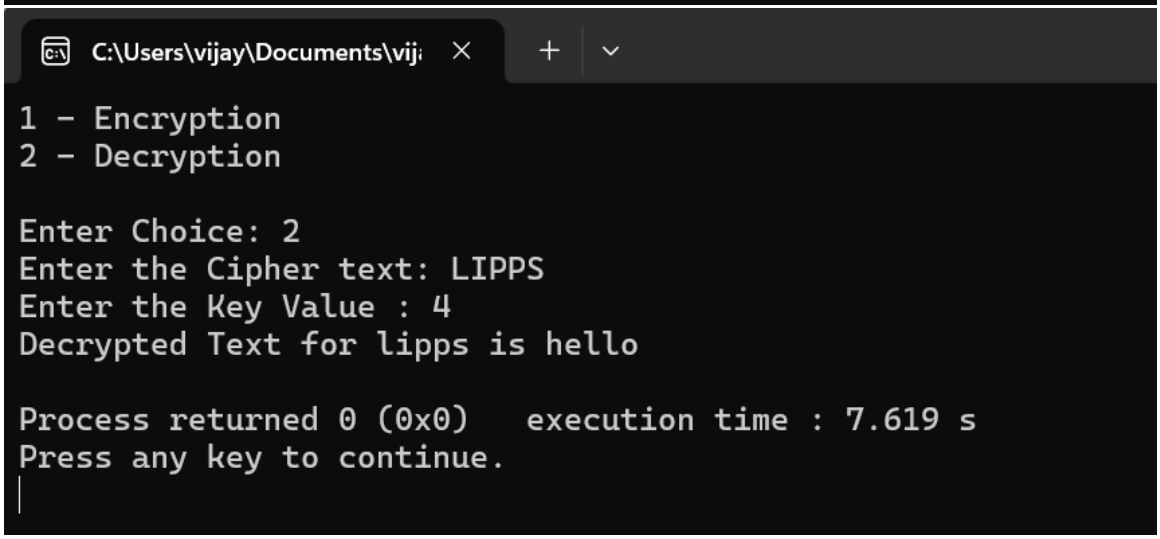
```c
    printf("Encrypted Text for %s is %s\n", p, c);
    }
    else if (choice==2)
    {
        printf("Enter the Cipher text: ");
        scanf("%s", p);
    for(i = 0; i<strlen(p); i++)
    {
        if(p[i]>='A' && p[i]<='Z')
        {
            p[i] = p[i] + 32;
        }
    }
    printf("Enter the Key Value : ");
    scanf("%d", &key);
    for (i = 0; i<strlen(p); i++)
    {
        if (p[i] >= 'a' && p[i] <= 'z')
        {
            c[i] = ((p[i] - key));
            if(c[i]<'a')
            {
                c[i]+=26;
            }
        }
    }
    c[i]='\0';
    printf("Decrypted Text for %s is %s\n", p, c);
    }
 else {
printf("Invalid choice\n");
}
return 0;
}
```

Output :


C:\Users\vijay\Documents\vij
1 - Encryption
2 - Decryption
Enter Choice: 1
Enter the Plaintext: drapjabdulkalam
Enter the Key Value : 4
Encrypted Text for drapjabdulkalam is HVETNEFHYPOEPEQ
Process returned 0 (0x0)   execution time : 14.135 s
Press any key to continue.


C:\Users\vijay\Documents\vij
1 - Encryption
2 - Decryption
Enter Choice: 2
Enter the Cipher text: HVETNEFHYPOEPEQ
Enter the Key Value : 4
Decrypted Text for hvetnefhypoepeq is drapjabdulkalam
Process returned 0 (0x0)   execution time : 8.489 s
Press any key to continue.





- Playfair cipher (Encryption only)

Code :

```
#include<stdio.h>
#include<string.h>
int main()
{
   char key[20], message[30];
   printf("Enter the key : ");
   scanf("%s", key);
   char playfair[5][5], pf[25];
   int l=strlen(key);
   int i, j, pfi=1, check, ci=0, cj=0, ck=0, cx=0, cy=0;
   for(i=0; i<25; i++)
   {
      pf[i]='0';
   }
   pf[0]=key[0];
   for(i=0; i<l; i++)
   {
      if(key[i]=='i')
      {
         ci=1;
      }
      if(key[i]=='x')
      {
```

```
        cx=1;
    }
    if(key[i]=='y')
    {
        cy=1;
    }
    if(key[i]=='j')
    {
        cj=1;
    }
    if(key[i]=='k')
    {
        ck=1;
    }
    check=0;
    for(j=0; j<pfi; j++)
    {
        if(key[i]==pf[j])
            {
                check=1;
            }
    }
    if(check==0)
    {
        pf[pfi]=key[i];
        pfi++;
    }
}
char filler[25]="abcdefghiklmnopqrstuvwxyz";
char fillerxy[25]="abcdefghijklmnopqrstuvwxz";
int ij=0, jk=0, ik=0, xy=0, xz=0, yz=0;
ij=1;
if (ci==1)
{
    filler[9]='j';
    ik=1;
    if (cj==1 || ck==1)
    {
        for(int k=0; k<25; k++)
        {
            filler[k]=fillerxy[k];
            xy=1;
            if(cy==1)
            {
                xz=1;
            }
            if (cx==0)
            {
                yz=1;
            }
        }
    }
}
else
{
    if (cj==1)
    {
        filler[9]='j';
        ik=1;
        if (ck==1)
        {
            for(int k=0; k<25; k++)
            {
                filler[k]=fillerxy[k];
                xy=1;
            }
        }
    }
}
for(i=0; i<25; i++)
{
    check=0;
    for(j=0; j<pfi; j++)
    {
        if(filler[i]==pf[j])
            {
                check=1;
            }
    }
    if(check==0)
    {
        pf[pfi]=filler[i];
        pfi++;
    }
}
pfi=0;
for(i=0; i<5; i++)
{
    for(j=0; j<5; j++)
    {
        playfair[i][j]=pf[pfi];
        pfi++;
    }
```

```
}
printf("Enter the Message without spaces : ");
scanf("%s", message);
l=strlen(message);
char me[l][2];
int count=0;
for(i=0; i<l; i++)
{
   if(count%2==1)
   {
      if (message[i]==me[count/2][0])
      {
         me[count/2][1]='x';
         count++;
      }
   }
   me[count/2][count%2]=message[i];
   count++;
}
char encrpt[l];
int encrpti=0;
if (count%2==1)
{
   me[count/2][count%2]='x';
}
char f1;
char f2;
for(i=0; i<(count+1)/2; i++)
{
   for(j=0; j<2; j++)
   {
      f1=me[i][j];
      if(ij==1 && f1=='j')
      {
         f1='i';
      }
      if(jk==1 && f1=='k')
      {
         f1='j';
      }
      if(ik==1 && f1=='k')
      {
         f1='i';
      }
      if(xy==1 && f1=='y')
      {
         f1='x';
      }
      if(xz==1 && f1=='z')
      {
         f1='x';
      }
      if(yz==1 && f1=='z')
      {
         f1='y';
      }
      me[i][j]=f1;
   }
}
int f1i, f1j, f2i, f2j;
l=count;
for(i=0; i<((count+1)/2); i++)
{
   f1=me[i][0];
   f2=me[i][1];
   for(j=0; j<5; j++)
   {
      for(int k=0; k<5; k++)
      {
         if(f1==playfair[j][k])
         {
            f1i=j;
            f1j=k;
         }
         if(f2==playfair[j][k])
         {
            f2i=j;
            f2j=k;
         }
      }
   }
   if (f1i==f2i)
   {
      encrpt[encrpti]=playfair[f1i][f1j+1];
      encrpti++;
      encrpt[encrpti]=playfair[f1i][f2j+1];
      encrpti++;
   }
   else if (f1j==f2j)
   {
      encrpt[encrpti]=playfair[f1i+1][f2j];
      encrpti++;
      encrpt[encrpti]=playfair[f2i+1][f2j];
```

```
        encrpti++;
    }
    else
    {
        encrpt[encrpti]=playfair[f1i][f2j];
        encrpti++;
        encrpt[encrpti]=playfair[f2i][f1j];
        encrpti++;
    }
}
for(i=0; i<l; i++)
{
    printf("%c", encrpt[i]);
}
}
```

Output:

- Vignere cipher

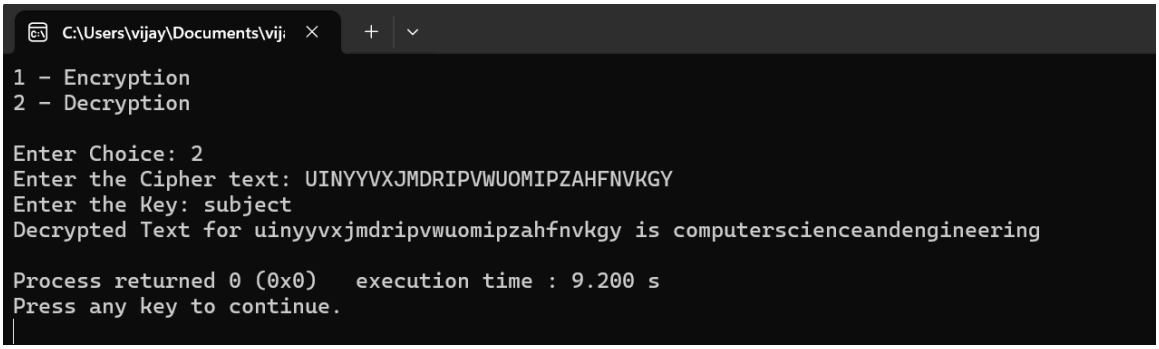
Code:

Start here | vijaycrypto.c

```c
#include <stdio.h>
#include <string.h>
#include <ctype.h>

int main() {
    char p[100], c[100], key[100];
    int i, choice;
    printf("1 - Encryption\n2 - Decryption\n\nEnter Choice: ");
    scanf("%d", &choice);
    if (choice == 1) {
        printf("Enter the Plaintext: ");
        scanf("%s", p);
        for (i = 0; i < strlen(p); i++) {
            if (p[i] >= 'A' && p[i] <= 'Z') {
                p[i] = p[i] + 32;
            }
        }
        printf("Enter the Key: ");
        scanf("%s", key);
        int key_len = strlen(key);
        for (i = 0; i < key_len; i++) {
            if (key[i] >= 'A' && key[i] <= 'Z') {
                key[i] = key[i] + 32;
            }
        }
        for (i = 0; i < strlen(p); i++) {
            if (p[i] >= 'a' && p[i] <= 'z') {
                c[i] = ((p[i] - 'a') + (key[i % key_len] - 'a')) % 26 + 'A';
            }
        }
        c[i] = '\0';
        printf("Encrypted Text for %s is %s\n", p, c);
    } else if (choice == 2) {
        printf("Enter the Cipher text: ");
        scanf("%s", p);
        for (i = 0; i < strlen(p); i++) {
            if (p[i] >= 'A' && p[i] <= 'Z') {
                p[i] = p[i] + 32;
            }
        }
        printf("Enter the Key: ");
        scanf("%s", key);
        int key_len = strlen(key);
        for (i = 0; i < key_len; i++) {
            if (key[i] >= 'A' && key[i] <= 'Z') {
                key[i] = key[i] + 32;
            }
        }
        for (i = 0; i < strlen(p); i++) {
            if (p[i] >= 'a' && p[i] <= 'z') {
                c[i] = ((p[i] - 'a') - (key[i % key_len] - 'a') + 26) % 26 + 'a';
            }
        }
        c[i] = '\0';
        printf("Decrypted Text for %s is %s\n", p, c);
    } else {
        printf("Invalid choice\n");
    }
    return 0;
}
```

Output :

```
1 - Encryption
2 - Decryption

Enter Choice: 1
Enter the Plaintext: ComputerScienceandEngineering
Enter the Key: subject
Encrypted Text for computerscienceandengineering is UINYYVXJMDRIPVWUOMIPZAHFNVKGY

Process returned 0 (0x0)   execution time : 116.811 s
Press any key to continue.
```



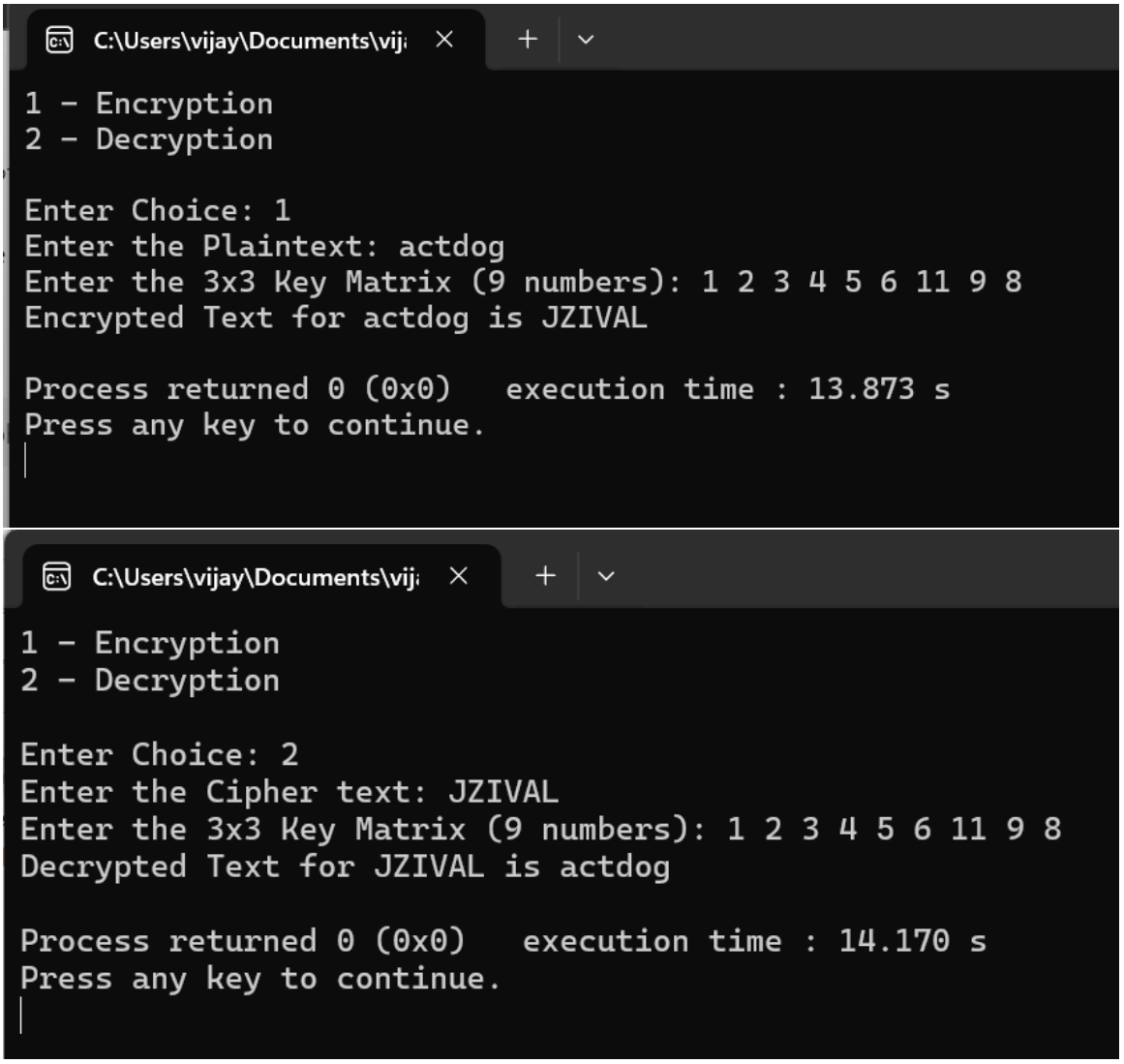
- Hill cipher

Code :

```
#include <stdio.h>
#include <string.h>

void encrypt(char* p, int key[3][3], char* c) {
    int i, j, k;
    int temp[3];

    for (i = 0; i < strlen(p); i += 3) {
        for (j = 0; j < 3; j++) {
            temp[j] = 0;
            for (k = 0; k < 3; k++) {
                temp[j] += (p[i + k] - 'a') * key[k][j];
            }
            temp[j] %= 26;
        }
        for (j = 0; j < 3; j++) {
            c[i + j] = temp[j] + 'A';
        }
    }
    c[i] = '\0';
}

void decrypt(char* p, int key[3][3], char* c) {
    int i, j, k;
    int temp[3];
    int inv[3][3];
    int det = key[0][0] * (key[1][1] * key[2][2] - key[1][2] * key[2][1]) -
              key[0][1] * (key[1][0] * key[2][2] - key[1][2] * key[2][0]) +
              key[0][2] * (key[1][0] * key[2][1] - key[1][1] * key[2][0]);
    det = (det % 26 + 26) % 26;

    int inv_det = -1;
    for (i = 0; i < 26; i++) {
        if ((det * i) % 26 == 1) {
            inv_det = i;
            break;
        }
    }

    inv[0][0] = (key[1][1] * key[2][2] - key[1][2] * key[2][1]) * inv_det % 26;
    inv[0][1] = (key[0][2] * key[2][1] - key[0][1] * key[2][2]) * inv_det % 26;
    inv[0][2] = (key[0][1] * key[1][2] - key[0][2] * key[1][1]) * inv_det % 26;
    inv[1][0] = (key[1][2] * key[2][0] - key[1][0] * key[2][2]) * inv_det % 26;
    inv[1][1] = (key[0][0] * key[2][2] - key[0][2] * key[2][0]) * inv_det % 26;
    inv[1][2] = (key[0][2] * key[1][0] - key[0][0] * key[1][2]) * inv_det % 26;
    inv[2][0] = (key[1][0] * key[2][1] - key[1][1] * key[2][0]) * inv_det % 26;
    inv[2][1] = (key[0][1] * key[2][0] - key[0][0] * key[2][1]) * inv_det % 26;
    inv[2][2] = (key[0][0] * key[1][1] - key[0][1] * key[1][0]) * inv_det % 26;

    for (i = 0; i < 3; i++) {
        for (j = 0; j < 3; j++) {
            if (inv[i][j] < 0) {
                inv[i][j] += 26;
            }
        }
    }

    for (i = 0; i < strlen(p); i += 3) {
        for (j = 0; j < 3; j++) {
            temp[j] = 0;
            for (k = 0; k < 3; k++) {
```

```
            temp[j] += (p[i + k] - 'A') * inv[k][j];
        }
        temp[j] %= 26;
    }
    for (j = 0; j < 3; j++) {
        c[i + j] = temp[j] + 'a';
    }
  }
  c[i] = '\0';
}

int main() {
  char p[100], c[100];
  int key[3][3], i, choice;
  printf("1 - Encryption\n2 - Decryption\n\nEnter Choice: ");
  scanf("%d", &choice);
  if (choice == 1) {
    printf("Enter the Plaintext: ");
    scanf("%s", p);
    for (i = 0; i < strlen(p); i++) {
      if (p[i] >= 'A' && p[i] <= 'Z') {
        p[i] = p[i] + 32;
      }
    }
    printf("Enter the 3x3 Key Matrix (9 numbers): ");
    scanf("%d %d %d %d %d %d %d %d %d", &key[0][0], &key[0][1], &key[0][2], &key[1][0], &key[1][1], &key[1][2], &key[2][0], &key[2][1],
&key[2][2]);
    encrypt(p, key, c);
    printf("Encrypted Text for %s is %s\n", p, c);
  } else if (choice == 2) {
    printf("Enter the Cipher text: ");
    scanf("%s", p);
    printf("Enter the 3x3 Key Matrix (9 numbers): ");
    scanf("%d %d %d %d %d %d %d %d %d", &key[0][0], &key[0][1], &key[0][2], &key[1][0], &key[1][1], &key[1][2], &key[2][0], &key[2][1],
&key[2][2]);
    decrypt(p, key, c);
    printf("Decrypted Text for %s is %s\n", p, c);
  } else {
    printf("Invalid choice\n");
  }
  return 0;
}
```

OUTPUT:


C:\Users\vijay\Documents\vij
1 - Encryption
2 - Decryption
Enter Choice: 1
Enter the Plaintext: actdog
Enter the 3x3 Key Matrix (9 numbers): 1 2 3 4 5 6 11 9 8
Encrypted Text for actdog is JZIVAL
Process returned 0 (0x0) execution time : 13.873 s
Press any key to continue.

C:\Users\vijay\Documents\vij
1 - Encryption
2 - Decryption
Enter Choice: 2
Enter the Cipher text: JZIVAL
Enter the 3x3 Key Matrix (9 numbers): 1 2 3 4 5 6 11 9 8
Decrypted Text for JZIVAL is actdog
Process returned 0 (0x0) execution time : 14.170 s
Press any key to continue.