LAB ASSESSMENT 1

Name: Charan Lalchand Soneji

Registration number: 17BCE2196

Faculty: Prof Marimuthu K.

Course Title: Cryptography fundamentals

Slot: L51+ L52

1. **To implement Ceaser Cipher technique**
2. **To implement Playfair Cipher technique**

Code

```
#include<stdio.h>
#include<math.h>
#include<conio.h>
#include<string.h>
#include <ctype.h>
#define MX 5
int choice;

void encryption1(){
     char message[100];
     char charac;
     int i, key;

     printf("Enter a message to encrypt:\n");
     scanf("%s",message);
     printf("Enter key: ");
     scanf("%d", &key);

     for(i = 0; message[i] != '\0'; ++i){
          charac = message[i];

          if(charac >= 'a' && charac <= 'z'){
               charac = charac + key;

               if(charac > 'z'){
                    charac = charac - 'z' + 'a' - 1;
               }

               message[i] = charac;
          }
          else if(charac >= 'A' && charac <= 'Z'){
               charac = charac + key;
```

```
                if(charac > 'Z'){
                    charac = charac - 'Z' + 'A' - 1;
                }

                message[i] = charac;
            }
    }
        printf("Encrypted message: %s\n", message);
    }
    void decryption1(){
        char message[100];
        char charac;
        int i, key;

        printf("Enter a message to decrypt: ");
        scanf("%s",message);
        printf("Enter key: ");
        scanf("%d", &key);

        for(i = 0; message[i] != '\0'; ++i){
            charac = message[i];

            if(charac >= 'a' && charac <= 'z'){
                charac = charac - key;

                if(charac < 'a'){
                    charac = charac + 'z' - 'a' + 1;
                }

                message[i] = charac;
            }
            else if(charac >= 'A' && charac <= 'Z'){
                charac = charac - key;

                if(charac < 'A'){
                    charac = charac + 'Z' - 'A' + 1;
                }

                message[i] = charac;
            }
        }

        printf("Decrypted message: %s\n", message);
    }

    void playfair(char ch1, char ch2, char key[MX][MX]) {
        int i, j, w, x, y, z;
        for (i = 0; i < MX; i++) {
```

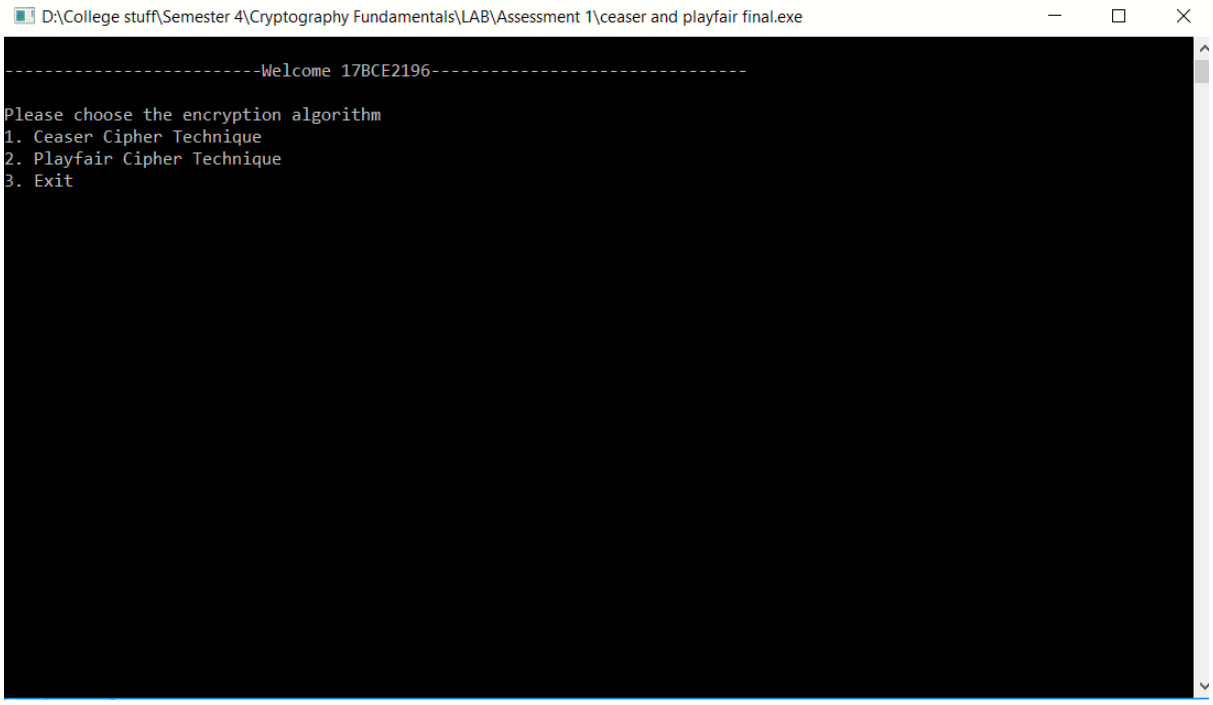
```
            for (j = 0; j < MX; j++) {
                if (ch1 == key[i][j]) {
                    w = i;
                    x = j;
                } else if (ch2 == key[i][j]) {
                    y = i;
                    z = j;
                }
            }
        }
        //printf("%d%d %d%d",w,x,y,z);
        if (w == y) {
              if(choice==1){
                   x = (x + 1) % 5;
                   z = (z + 1) % 5;
              }
              else{
                   x = ((x - 1) % 5+5)%5;
                   z = ((z - 1) % 5+5)%5;
              }
            printf("%c%c", key[w][x], key[y][z]);
        } else if (x == z) {
              if(choice==1){
                   w = (w + 1) % 5;
                   y = (y + 1) % 5;
              }
              else{
                   w = ((w - 1) % 5+5)%5;
                   y = ((y - 1) % 5+5)%5;
              }
            printf("%c%c", key[w][x], key[y][z]);
        }
         else {
            printf("%c%c", key[w][z], key[y][x]);
        }
    }
    void removeDuplicates(char str[]){
        int hash[256]        =  {0};
        int currentIndex     = 0;
        int lastUniqueIndex  = 0;
        while(*(str+currentIndex)){
            char temp = *(str+currentIndex);
            if(0 == hash[temp]){
                hash[temp] = 1;
                *(str+lastUniqueIndex) = temp;
                lastUniqueIndex++;
            }
            currentIndex++;
```

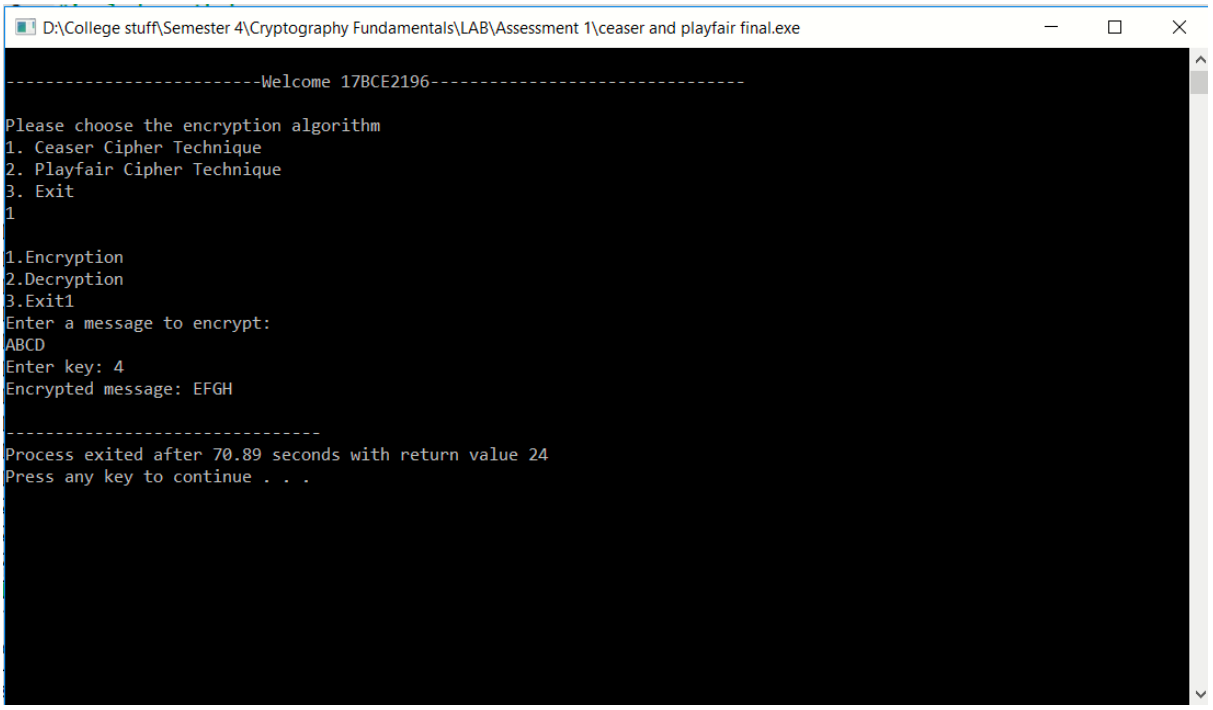
```
    }
    *(str+lastUniqueIndex) = '\0';

}

void playfairfinal(){
     int i, j, k = 0, l, m = 0, n;
    char key[MX][MX], keyminus[25], keystr[10], str[25] =
{
        0
    };

    char alpa[26] = {

'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'
    };
     printf("\n1.Encryption\n2.Decryption\n3.Exit");
    scanf("%d",&choice);
     if(choice!=1 && choice!=2 && choice!=3){
printf("Invalid Choice"); return;}
     fflush(stdin);
    printf("\nEnter key:");
    gets(keystr);
    printf("Enter the message:");
    gets(str);
     removeDuplicates(keystr);
    n = strlen(keystr);
    //convert the characters to uppertext
    for (i = 0; i < n; i++) {
        if (keystr[i] == 'j') keystr[i] = 'i';
        else if (keystr[i] == 'J') keystr[i] = 'I';
        keystr[i] = toupper(keystr[i]);
    }
    //convert all the characters of plaintext to
uppertext
    for (i = 0; i < strlen(str); i++) {
        if (str[i] == 'j') str[i] = 'i';
        else if (str[i] == 'J') str[i] = 'I';
        str[i] = toupper(str[i]);
    }
    // store all characters except key
    j = 0;
    for (i = 0; i < 26; i++) {
        for (k = 0; k < n; k++) {
            if (keystr[k] == alpa[i]) break;
            else if (alpa[i] == 'J') break;
        }
```

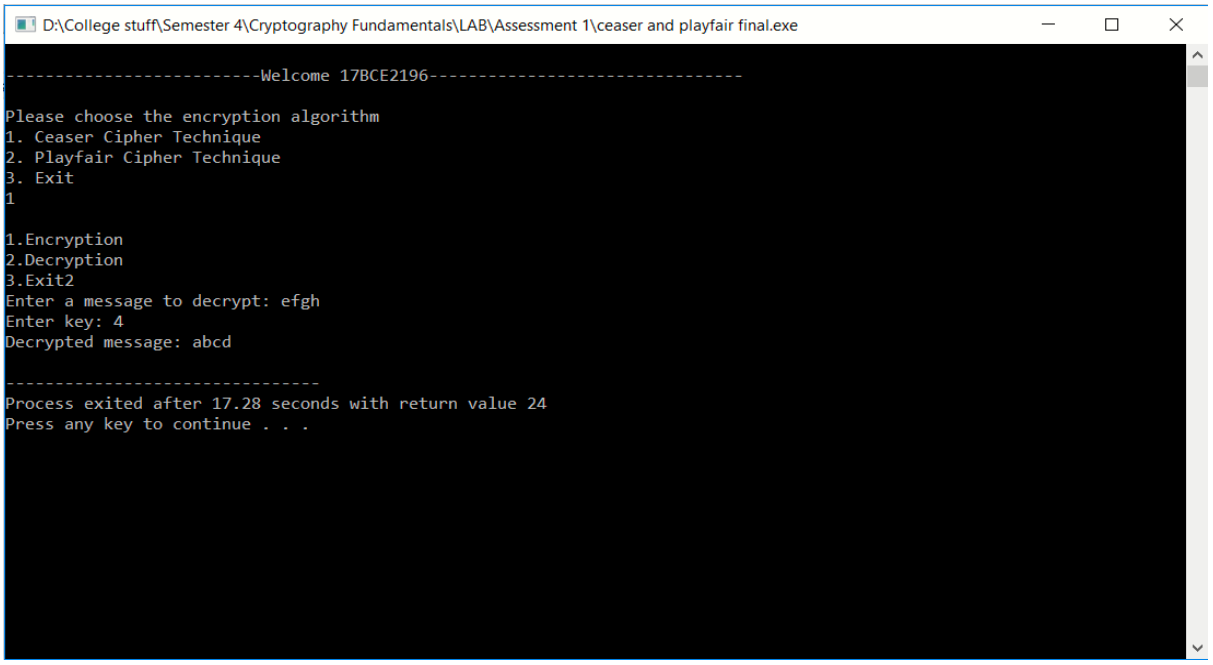
```
        if (k == n) {
            keyminus[j] = alpa[i];
            j++;
        }
    }
    //construct key keymatrix
    k = 0;
    for (i = 0; i < MX; i++) {
        for (j = 0; j < MX; j++) {
            if (k < n) {
                key[i][j] = keystr[k];
                k++;
            } else {
                key[i][j] = keyminus[m];
                m++;
            }
            printf("%c ", key[i][j]);
        }
        printf("\n");
    }
    // construct diagram and convert to cipher text
    printf("\nEntered text :%s\nOutput Text :", str);
    for (i = 0; i < strlen(str); i++) {
        if (str[i] == 'J') str[i] = 'I';
        if (str[i + 1] == '\0') playfair(str[i], 'X',
key);
        else {
            if (str[i + 1] == 'J') str[i + 1] = 'I';
            if (str[i] == str[i + 1]) playfair(str[i],
'X', key);
            else {
                playfair(str[i], str[i + 1], key);
                i++;
            }
        }
    }
     if(choice==2) printf(" (Remove unnecessary X)");

}


int main(){
     printf("\n-------------------------Welcome
17BCE2196-------------------------------\n");
     int o;
     printf("\nPlease choose the encryption algorithm");
     printf("\n1. Ceaser Cipher Technique");
     printf("\n2. Playfair Cipher Technique");
```

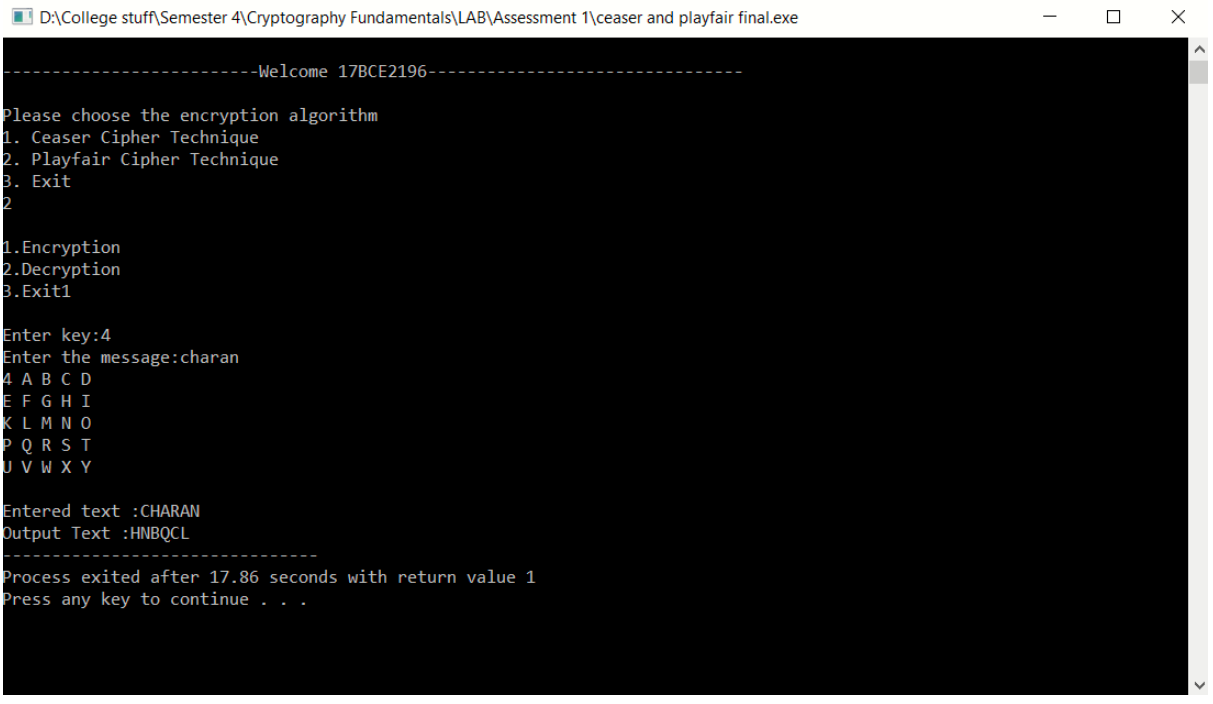
```
        printf("\n3. Exit\n");
        scanf("\n%d",&o);
        switch(o){
             case 1:

        printf("\n1.Encryption\n2.Decryption\n3.Exit");
             scanf("%d",&choice);
                  if(choice!=1 && choice!=2 && choice!=3){
    printf("Invalid Choice"); return 0;}
                  switch(choice){
                       case 1:
                            encryption1();
                            break;
                       case 2:
                            decryption1();
                            break;
                       case 3:
                            break;
                  }
                  break;

             case 2:
                  playfairfinal();
                  break;
             case 3:
                  break;
        }

    }
```

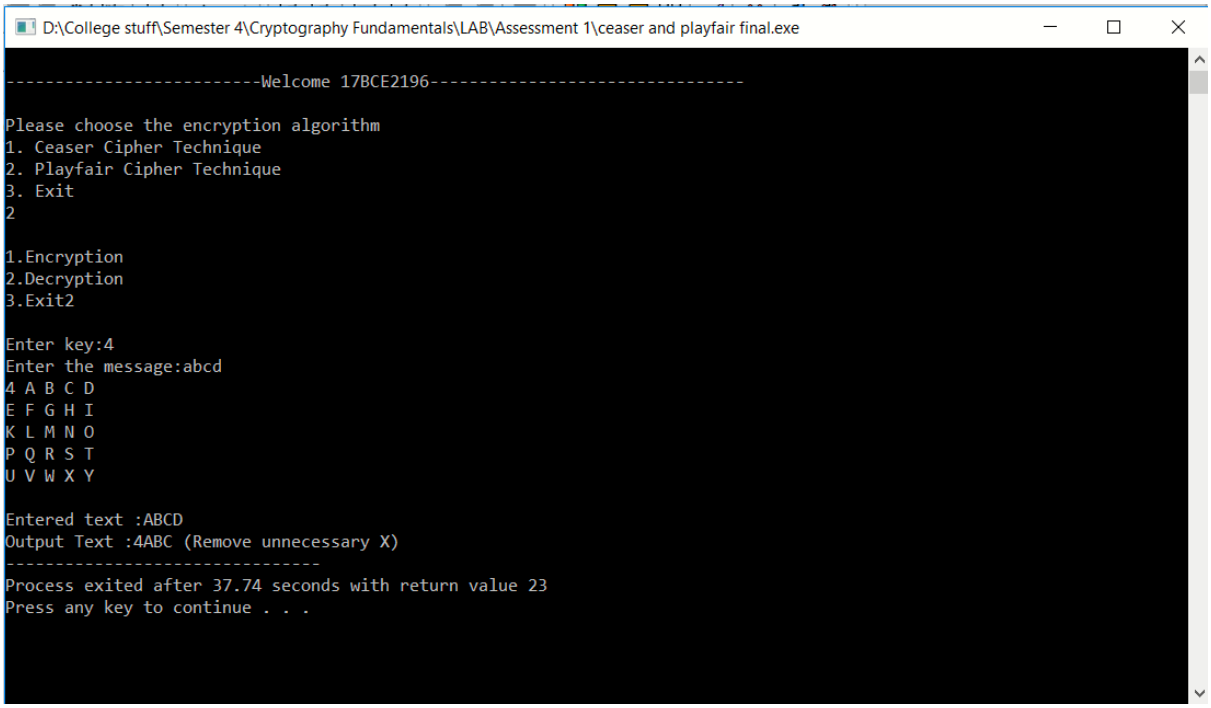
OUTPUT SNIPPET

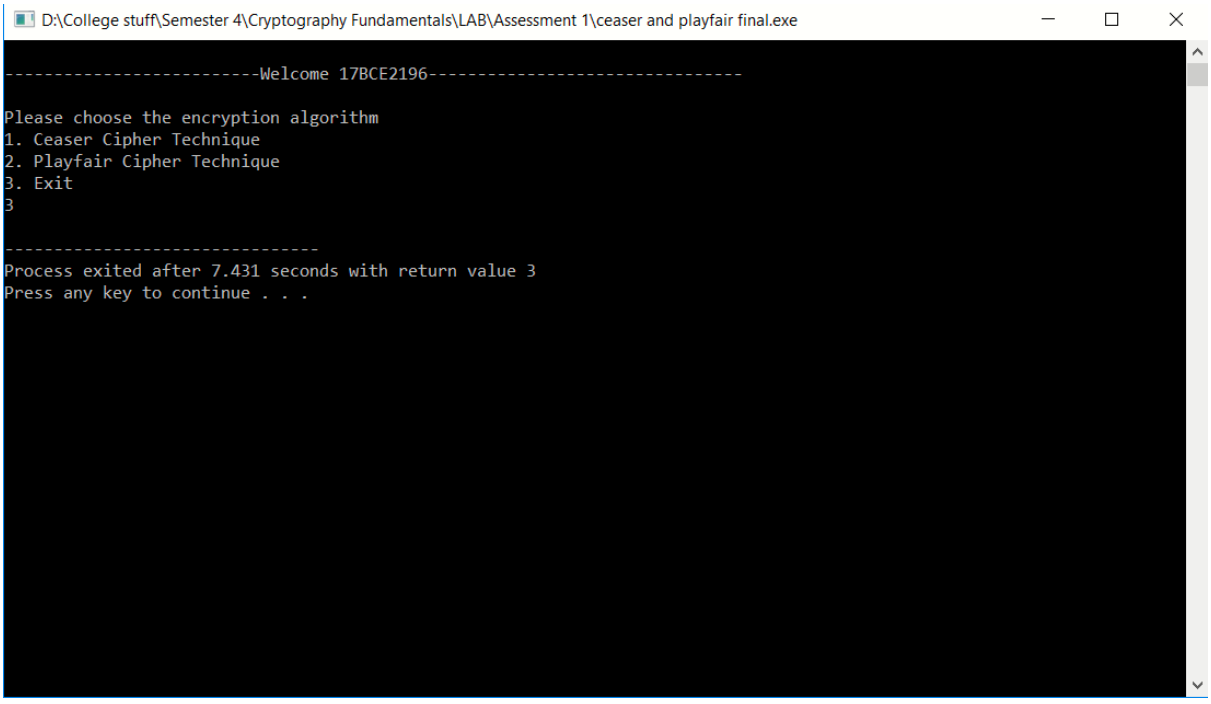



D:\College stuff\Semester 4\Cryptography Fundamentals\LAB\Assessment 1\ceaser and playfair final.exe
----------Welcome 17BCE2196----------
Please choose the encryption algorithm
1. Ceaser Cipher Technique
2. Playfair Cipher Technique
3. Exit
1
1.Encryption
2.Decryption
3.Exit1
Enter a message to encrypt:
ABCD
Enter key: 4
Encrypted message: EFGH
Process exited after 70.89 seconds with return value 24
Press any key to continue . . .


D:\College stuff\Semester 4\Cryptography Fundamentals\LAB\Assessment 1\ceaser and playfair final.exe
----------Welcome 17BCE2196----------
Please choose the encryption algorithm
1. Ceaser Cipher Technique
2. Playfair Cipher Technique
3. Exit
1
1.Encryption
2.Decryption
3.Exit2
Enter a message to decrypt: efgh
Enter key: 4
Decrypted message: abcd
Process exited after 17.28 seconds with return value 24
Press any key to continue . . .


D:\College stuff\Semester 4\Cryptography Fundamentals\LAB\Assessment 1\ceaser and playfair final.exe
----------Welcome 17BCE2196----------
Please choose the encryption algorithm
1. Ceaser Cipher Technique
2. Playfair Cipher Technique
3. Exit
2
1.Encryption
2.Decryption
3.Exit1
Enter key:4
Enter the message:charan
4 A B C D
E F G H I
K L M N O
P Q R S T
U V W X Y
Entered text :CHARAN
Output Text :HNBQCL
Process exited after 17.86 seconds with return value 1
Press any key to continue . . .


D:\College stuff\Semester 4\Cryptography Fundamentals\LAB\Assessment 1\ceaser and playfair final.exe
----------Welcome 17BCE2196----------
Please choose the encryption algorithm
1. Ceaser Cipher Technique
2. Playfair Cipher Technique
3. Exit
2
1.Encryption
2.Decryption
3.Exit2
Enter key:4
Enter the message:abcd
4 A B C D
E F G H I
K L M N O
P Q R S T
U V W X Y
Entered text :ABCD
Output Text :4ABC (Remove unnecessary X)
Process exited after 37.74 seconds with return value 23
Press any key to continue . . .


D:\College stuff\Semester 4\Cryptography Fundamentals\LAB\Assessment 1\ceaser and playfair final.exe
----------Welcome 17BCE2196----------
Please choose the encryption algorithm
1. Ceaser Cipher Technique
2. Playfair Cipher Technique
3. Exit
3
Process exited after 7.431 seconds with return value 3
Press any key to continue . . .