Name : Janisha Singh
Course : BSc - IT (6th Sem)
Roll no : 1022771

Subject : Information Security

Q5

```
#include <stdio.h>
#include <string.h>
#include <ctype.h>
int removerepeated(int size, int a[]);
int insertelementat(int position, int a[],
int size);
int main()
{
  int i, j, K, numstr[100], numcipher[100],
numKey[100], lenKey, templen, tempKey[100],
flag=-1, size, cipherKey[5][5], lennumstr, row1,
row2, col1, col2;
char str[100], Key[100];
printf("Enter string");
gets(str);
for(i=0; j=0; i<strlen(str); i++)
{
  if(str[i]!=' ') {
    str[j]=toupper(str[i]);
    j++; }
str[j]='\0';
printf("Entered String is %s", str);
size=strlen(str);
```

Janisha

```
for (int i=0; i<size; i++)
{ if (str[i] != ' ')
    numstr[i] = str[i] - 'A';
  }
  lennumstr = i;
printf ("Enter Key \n"); gets (Key);
for (i=0; j=0; i<strlen(Key); i++)
{
  if (Key[i] != '') {
    Key[j] = toupper (Key [i]);
j++;
}
 Key [j] = '\0';
printf ("%s\n", Key);
 K=0;
for (i=0; i<strlen (Key) + 26; i++)
 {
  if (i<strlen (Key))
  {
    if (Key [i] == 'J')
  {
   flag = 8;
    printf ("%d", flag);
  }
  numKey[i] = Key[i] - 'A';
}
else
```

Tanisha

```
{
  if (K != 9 && K != flag)
  {
    numKey[i] = K;
  }
  K++;
}
templen = i;
lenKey = removerepeated(templen, numKey);
printf("Entered Key converted \n");
for(i=0; i<lenKey; i++)
{
  printf("%c", numKey[i] + 'A');
}
K=0;
for(i=0; i<5; i++)
{
  for(j=0; j<5; j++)
  {
    cipherKey[i][j] = numKey[K];
    K++;
  }
}
printf("Arranged Key\n");
for(i=0; i<5; i++)
{
  for(j=0; j<5; j++)
    printf("%c", cipherKey[i][j] + 'A');
}
```

Vanishing

```
for(i=0; i<lennumstr; i+=2)
{
  if(numstr[i]==numstr[i+1])
  {
  insertelementat(i+1, numstr, lennumstr);
  lennumstr++;
  }
}
if(lennumstr%2!=0)
{
  insertelementat(lennumstr, numstr, lennum
                                    -str);
  lennumstr++;
}
printf("Entered String ");
for(i=0; i<lennumstr; i++)
printf("%c", numstr[i]+'A');
{
for(K=0; K<lennumstr; K+=2)
{
  for(i=0; i<5; i++)
    for(j=0; j<5; j++)
  {
  if(numstr[K]==cipherKey[i][j])
  {
    *row1=i;
     col1=j;
  }
```

Tanishq

```
if (numstr[K+1] = cipherKey[i][j])
{
    row2 = i;
    col2 = j;
}}}
if (row1 == row2)
col1 = (col1 + 1) % 5;
col2 = (col2+1) % 5;
if (col1 < 0)
{
    col1 = 5+ col1;
}
if (col2 < 0)
{
    col2 = 5+col2;
}
numcipher[K] = cipherKey[row1][col1];
numcipher[K+1] = cipherKey[row2][col2];
}
if (col1 == col2)
{
    row1 = (row1+1) % 5;
row2 = (row2 +1) % 5;
if (row1 < 0)
{
    row1 = 5+ row1;
}
    if (row2 < 0)
```

Tanisha

```
row2 = 5+row2;
}
numcipher[K] = cipherKey[row1][col1];
numcipher[K+1] = cipherKey[row2][col2];
}
if (row1 != row2 && col1 != col2)
{
numcipher[K] = cipherKey[row1][col2];
numcipher[K+1] = cipherKey[row2][col1];
}
}
printf("Cipher Text");
for (i=0; i<lenumstr; i++)
{
if (numcipher[i]+'A') != 'X').
printf("%c", numcipher[i]+'A');
}
printf("\n");
}
}
```

Tanisha

```
Enter a string
go with the flow
Entered String is GOWITHTHEFLOW
Enter the key (Non repeated elements if possible)
key
KEY
Entered key converted according to Play Fair Cipher rule
KEYABCDFGHI‖LMNOPQRSTUVWXZ
Arranged key
K E Y A B
C D F G H
I ‖ L M N
O P Q R S
T U V W X
Entered String/Message After Processing according to Play fair cipher rule
GOWITHTHEFLOWX
Cipher Text is
CRTMCCYDIQT

--------------------------------
Process exited after 10.69 seconds with return value 0
Press any key to continue . . .
```