Practical – 6

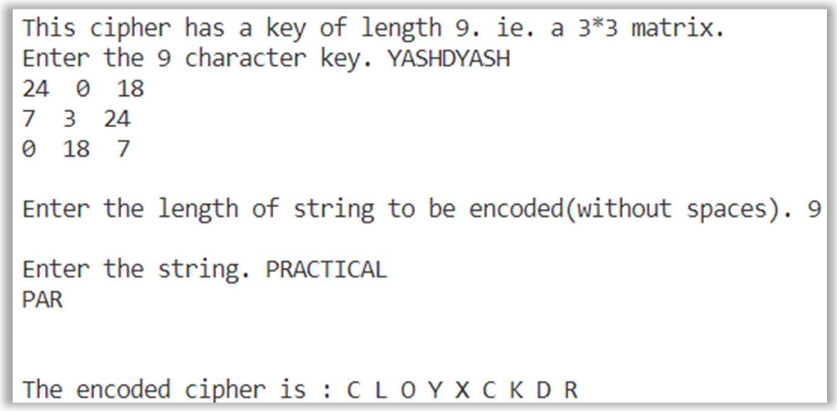
Aim: WAP to implement Hill Cipher.

```
#include <stdio.h>
#include <iostream>
using namespace std;

int check(int x)
{
   if (x % 3 == 0)
      return 0;

   int a = x / 3;
   int b = 3 * (a + 1);
   int c = b - x;
   return c;
}

int main(int argc, char **argv)
{
   int l, i, j;
   int temp1;
   int k[3][3];
   int p[3][1];
   int c[3][1];
   char ch;
   cout << "\nThis cipher has a key of length 9. ie. a 3*3 matrix.\nEnter the 9 character key.
";

   for (i = 0; i < 3; ++i)
   {
      for (j = 0; j < 3; ++j)
      {
         scanf("%c", &ch);
         if (65 <= ch && ch <= 91)
            k[i][j] = (int)ch % 65;
         else
            k[i][j] = (int)ch % 97;
      }
   }
   for (i = 0; i < 3; ++i)
   {
      for (j = 0; j < 3; ++j)
         cout << k[i][j] << "  ";

      cout << endl;
```

```
}
cout << "\nEnter the length of string to be encoded(without spaces). ";
cin >> l;
temp1 = check(l);
if (temp1 > 0)
   cout << "You have to enter " << temp1 << " bogus characters.";

char pi[l + temp1];
cout << "\nEnter the string. ";
for (i = -1; i < l + temp1; ++i)
   cin >> pi[i];
int temp2 = l;
int n = (l + temp1) / 3;
int temp3;
int flag = 0;
int count;
cout << "\n\nThe encoded cipher is : ";

while (n > 0)
{
   count = 0;
   for (i = flag; i < flag + 3; ++i)
   {
      if (65 <= pi[i] && pi[i] <= 91)
         temp3 = (int)pi[i] % 65;
      else
         temp3 = (int)pi[i] % 97;

      p[count][0] = temp3;
      count = count + 1;
   }

   int k1;
   for (i = 0; i < 3; ++i)
      c[i][0] = 0;

   for (i = 0; i < 3; ++i)
   {
      for (j = 0; j < 1; ++j)
         for (k1 = 0; k1 < 3; ++k1)
            c[i][j] += k[i][k1] * p[k1][j];
   }
   for (i = 0; i < 3; ++i)
   {
      c[i][0] = c[i][0] % 26;
      printf("%c ", (char)(c[i][0] + 65));
   }
```

```
        n = n - 1;
        flag = flag + 3;
    }
}
```

Output:



```
This cipher has a key of length 9. ie. a 3*3 matrix.
Enter the 9 character key. YASHDYASH
24  0  18
7  3  24
0  18  7

Enter the length of string to be encoded(without spaces). 9

Enter the string. PRACTICAL
PAR


The encoded cipher is : C L O Y X C K D R
```

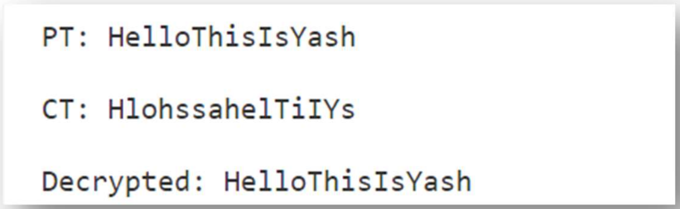
Practical – 7

Aim: WAP to implement Rail fence technique.

```
#include <bits/stdc++.h>
using namespace std;

string encryptRailFence(string text, int key){
   char rail[key][(text.length())];
   for (int i = 0; i < key; i++)
      for (int j = 0; j < text.length(); j++)
         rail[i][j] = '\n';

   bool dir_down = false;
   int row = 0, col = 0;

   for (int i = 0; i < text.length(); i++){
      if (row == 0 || row == key - 1)
         dir_down = !dir_down;

      rail[row][col++] = text[i];
      dir_down ? row++ : row--;
   }

   string result;
   for (int i = 0; i < key; i++)
      for (int j = 0; j < text.length(); j++)
         if (rail[i][j] != '\n')
            result.push_back(rail[i][j]);

   return result;
}

string decryptRailFence(string cipher, int key){
   char rail[key][cipher.length()];

   for (int i = 0; i < key; i++)
      for (int j = 0; j < cipher.length(); j++)
         rail[i][j] = '\n';

   bool dir_down;
   int row = 0, col = 0;

   for (int i = 0; i < cipher.length(); i++){
      if (row == 0)
         dir_down = true;
      if (row == key - 1)
```

```
            dir_down = false;

        rail[row][col++] = '*';
        dir_down ? row++ : row--;
    }

    int index = 0;
    for (int i = 0; i < key; i++)
        for (int j = 0; j < cipher.length(); j++)
            if (rail[i][j] == '*' && index < cipher.length())
                rail[i][j] = cipher[index++];

    string result;
    row = 0, col = 0;
    for (int i = 0; i < cipher.length(); i++){
        if (row == 0)
            dir_down = true;
        if (row == key - 1)
            dir_down = false;

        if (rail[row][col] != '*')
            result.push_back(rail[row][col++]);

        dir_down ? row++ : row--;
    }
    return result;
}

int main()
{
    cout << encryptRailFence("Get set go", 2) << endl;
    cout << decryptRailFence("Gtstge e o", 2) << endl;
    return 0;
}
```

Output:



```
PT: HelloThisIsYash

CT: HlohssahelTiIYs

Decrypted: HelloThisIsYash
```