**PRACTICAL - 04**

**Code:**

// Ashwin Navange A-38 CSE

#include<bits/stdc++.h>

using namespace std;

long int p, q, n, t, flag, e[100], d[100], temp[100], j, m[100], en[100], i;

char msg[100];

int prime(long int pr)

{

int i;

j = sqrt(pr);

for (i = 2; i <= j; i++)

{

if (pr % i == 0)

return 0;

}

return 1;

}

long int cd(long int x)

{

long int k = 1;

while (1)

{

k = k + t;

if (k % x == 0)

return (k / x);

}

}

void ce()

{

int k;

k = 0;

for (i = 2; i < t; i++)

{

if (t % i == 0)

continue;

flag = prime(i);

if (flag == 1 && i != p && i != q)

{

e[k] = i;

flag = cd(e[k]);

if (flag > 0)

{

d[k] = flag;

k++;

}

if (k == 99)

break;

}

}

}

void encrypt()

{

long int pt, ct, key = e[0], k, len;

i = 0;

len = strlen(msg);

while (i != len)

{

pt = m[i];

pt = pt - 96;

k = 1;

for (j = 0; j < key; j++)

{

k = k \* pt;

k = k % n;

}

temp[i] = k;

ct = k + 96;

en[i] = ct;

i++;

}

en[i] = -1;

cout << "\nTHE ENCRYPTED MESSAGE IS\n";

for (i = 0; en[i] != -1; i++)

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

}

void decrypt()

{

long int pt, ct, key = d[0], k;

i = 0;

while (en[i] != -1)

{

ct = temp[i];

k = 1;

for (j = 0; j < key; j++)

{

k = k \* ct;

k = k % n;

}

pt = k + 96;

m[i] = pt;

i++;

}

m[i] = -1;

cout << "\nTHE DECRYPTED MESSAGE IS\n";

for (i = 0; m[i] != -1; i++)

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

}

int main()

{

cout << "Ashwin Navange A-38 CSE\n";

cout << "\nEnter First Prime Number: ";

cin >> p;

flag = prime(p);

cout << "\nEnter Second Prime Number: ";

cin >> q;

cout << "\nEnter Plain Text: ";

fflush(stdin);

cin >> msg;

for (i = 0; msg[i] != '\0'; i++)

m[i] = msg[i];

n = p \* q;

t = (p - 1) \* (q - 1);

ce();

cout << "\nPossible Values Of e And d Are:\n";

for (i = 0; i < j - 1; i++)

cout << e[i] << "\t" << d[i] << "\n";

encrypt();

decrypt();

cout<<endl;

return 0;

}

**Output:**

