**KIET Group of Institutions ,Ghaziabad**

**COMPUTER SCIENCE AND INFORMATION TECHNOLOGY**



**PROJECT BASED LEARNING**

**On**

**CRPTOGRAPHY USING RSA ALGORITHM**

**SUBJECT: DISCRETE STRUCTURE AND LOGIC LAB**

**(KCS – 353)**

**Submitted By:**

ANUSHKA GUPTA – 2100290110033 (CSIT 3A)

AANYA GOEL – 2100290110002 (CSIT 3A)

**ACKNOWLEDGEMENT**

I’ve got this golden opportunity to express my kind gratitude and sincere thanks to my subject faculty “**MISS JAYA SHARMA**”, Computer Science and Information Technology Department, **KIET GROUP OF INSTITUTIONS** for their kind support and necessary counselling in the preparation of this project report. I’m also indebted to each and every person responsible for the making up of this project directly or indirectly.

I must also acknowledge or deep debt of gratitude each one of my colleague who led this project come out in the way it is. It’s my hard work and untiring sincere efforts and mutual cooperation to bring out the project work. Last but not the least, I would like to thank my parents for their sound counselling and cheerful support. They have always inspired us and kept our spirit up.

# INDEX

# INTRODUCTION

# REQUIREMENT ANALYSIS

# DESIGN

# CODING AND IMPLEMENTATION

# RESULT

# REFERENCE

**AIM**

Encrypted or encoded strings are most of the time used during transmission over the internet. By doing this it becomes a little bit safe to avoid the MITM (Man in the Middle) attacks and also prevent us from many more information leaks or data exposure.

**OBJECTIVE**

Encryption is the process by which a readable message is converted to an unreadable form to prevent unauthorized parties from reading it. Decryption is the process of converting an encrypted message back to its original (readable) format.

# ABSTRACT

The process of converting ordinary and plain text into unintelligible text and vice versa is known as cryptography. In this method, data is stored and transmitted in a specific form in order to make it available for only particular people to read and process.

RSA Algorithm is public-key cryptography which exposed from factorization attack based on the public key 'e' and modulus 'n'. The study aimed to modify an RSA Algorithm based on generation of different value of public key 'e' and show different result based on original equation of RSA Algorithm.

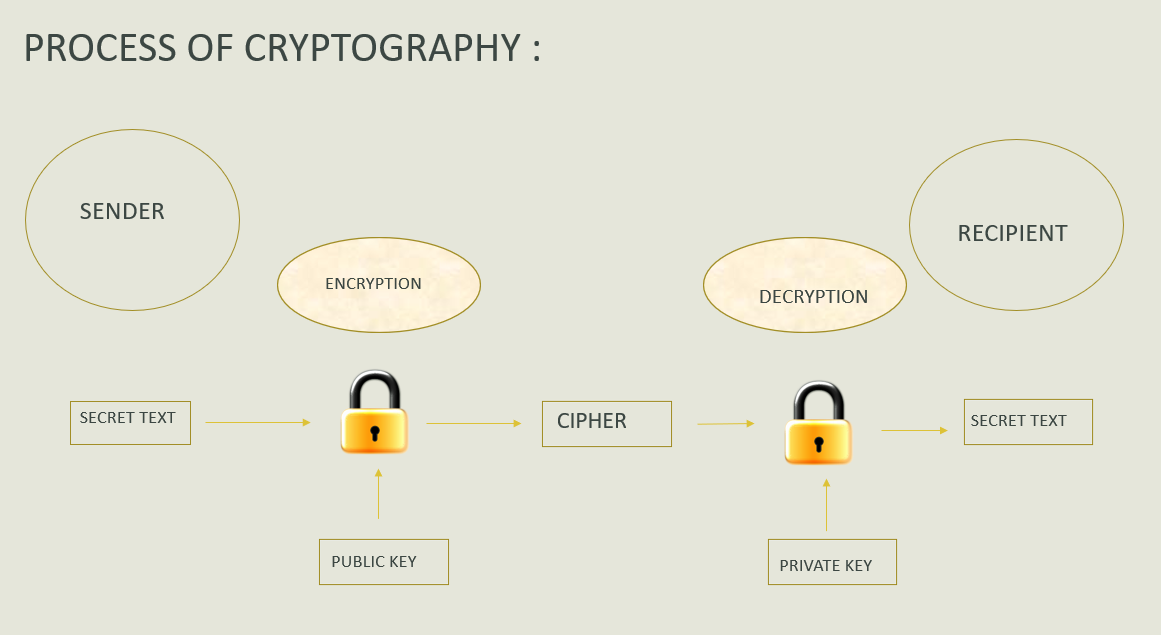
**BASIC PRINCIPLES:**

Applications of number theory allow the development of mathematical algorithms that can make information (data) unintelligible to everyone except for intended users. In addition, mathematical algorithms can provide real physical security to data—allowing only authorized users to delete or update data.

**USED OF DISCRETE MATHS IN CRYPTOGRAPHY**

The field of cryptography, which is the study of how to create security structures and passwords for computers and other electronic systems, is based entirely on discrete mathematics. This is partly because computers send information in discrete — or separate and distinct — bits. Number theory, one important part of discrete math, allows cryptographers to create and break numerical passwords. Because of the quantity of money and the amount of confidential information involved, cryptographers must first have a solid background in number theory to show they can provide secure passwords and encryption methods. Applications of number theory allow the development of mathematical algorithms that can make information (data) unintelligible to everyone except for intended users. In addition, mathematical algorithms can provide real physical security to data—allowing only authorized users to delete or update data.

**DIAGRAMATIC REPRESENTATIOIN OF CRYPTOGRAPGHY:**

****

**CODING IMPLEMENTATION**

#include<stdio.h>

#include<stdlib.h>

#include<math.h>

#include<string.h>

int x, y, n, t, i, flag;

long int e[50], d[50], temp[50], j, m[50], en[50];

char msg[100];

int prime(long int);

void encryption\_key();

long int cd(long int);

void encrypt();

void decrypt();

int main()

{

printf("\nENTER FIRST PRIME NUMBER\n");

scanf("%d", &x);

flag = prime(x);

if(flag == 0)

{

printf("\nINVALID INPUT\n");

exit(0);

}

printf("\nENTER SECOND PRIME NUMBER\n");

scanf("%d", &y);

flag = prime(y);

if(flag == 0 || x == y)

{

printf("\nINVALID INPUT\n");

exit(0);

}

printf("\nENTER MESSAGE OR STRING TO ENCRYPT\n");

scanf("%s",msg);

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

m[i] = msg[i];

n = x \* y;

t = (x-1) \* (y-1);

encryption\_key();

printf("\nPOSSIBLE VALUES OF e AND d ARE\n");

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

printf("\n%ld\t%ld", e[i], d[i]);

encrypt();

decrypt();

return 0;

}

int prime(long int pr)

{

int i;

j = sqrt(pr);

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

{

if(pr % i == 0)

return 0;

}

return 1;

}

//function to generate encryption key

void encryption\_key()

{

int k;

k = 0;

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

{

if(t % i == 0)

continue;

flag = prime(i);

if(flag == 1 && i != x && i != y)

{

e[k] = i;

flag = cd(e[k]);

if(flag > 0)

{

d[k] = flag;

k++;

}

if(k == 99)

break;

}

}

}

long int cd(long int a)

{

long int k = 1;

while(1)

{

k = k + t;

if(k % a == 0)

return(k / a);

}

}

//function to encrypt the message

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;

printf("\n\nTHE ENCRYPTED MESSAGE IS\n");

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

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

}

//function to decrypt the message

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;

printf("\n\nTHE DECRYPTED MESSAGE IS\n");

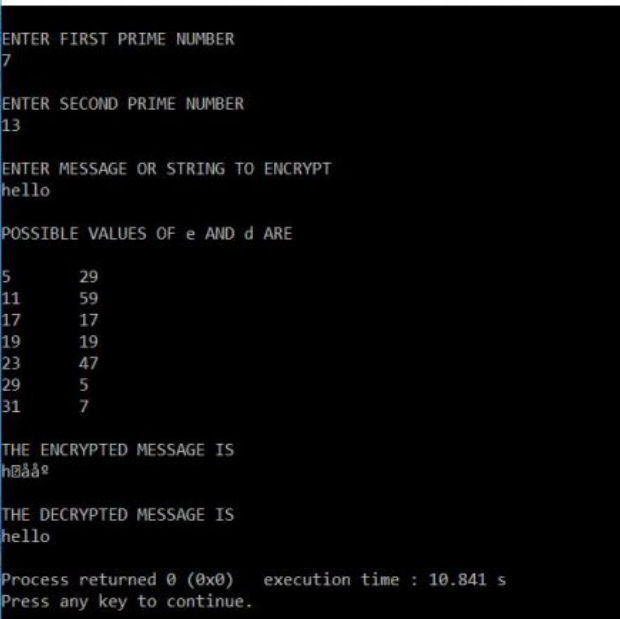
for(i = 0; m[i] != -1; i++)

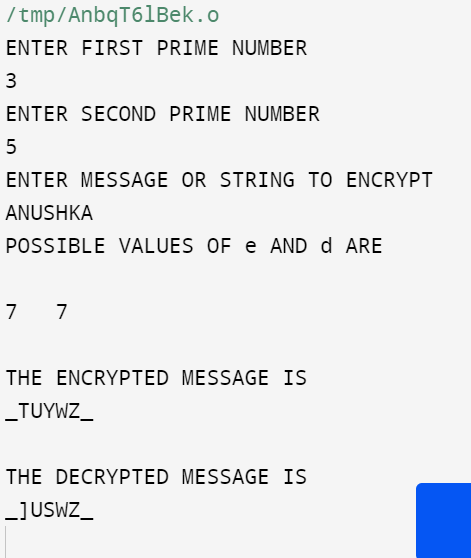
printf("%c", m[i]);

printf("\n");

}

**OUTPUT SCREENSHOT**





**GITHUB LINK:**

S

**REFERENCES:**

1. <http://www.trytoprogram.com/c-examples/c-program-to-encrypt-and-decrypt-string/#rsa>
2. <https://www.slideshare.net/talhasaleem09/cryptography-discrete-mathematics>