**PROJECT REPORT**

**ON**

ENCRYPTION OF DATA

****

REPORT SUBMITTED

TO

VISHWAKARMA INSTITUTE OF INFORMATION TECHNOLOGY, PUNE

FOR THE PBL OF COMPUTER FUNDAMENTALS

IN

**ENGINEERING AND APPLIED SCIENCE DEPARTMENT**

BY

**Kartikeya Vishnoi - 121010/ 21810071**

**Shreya Jadhav - 121006/ 21810101**

**Mirelle Martis - 121017/ 21810184**

**Parikshit Ranjeet Raje - 121021/ 21810234**

**Shubham Ubhe - 121015/ 21810164**

**Pragati Patil - 121020/ 21810231**

**Class: F.Y. B.TECH Division: COMP A Batch: A1**

**Batch Teacher: Prof. Ganesh C Shelke**

**INDEX**

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Contents** | **Page No.** |
| 1 | Abstract | 3 |
| 2 | Introduction and Theory | 4 |
| 3 | Flow of Program | 7 |
| 4 | Source code | 8 |
| 5 | Output | 16 |
| 6 | List of the topics / concept which are covered from the syllabus | 17 |
| 7 | Conclusion | 18 |
| 8 | References | 19 |

**ABSTRACT**

T

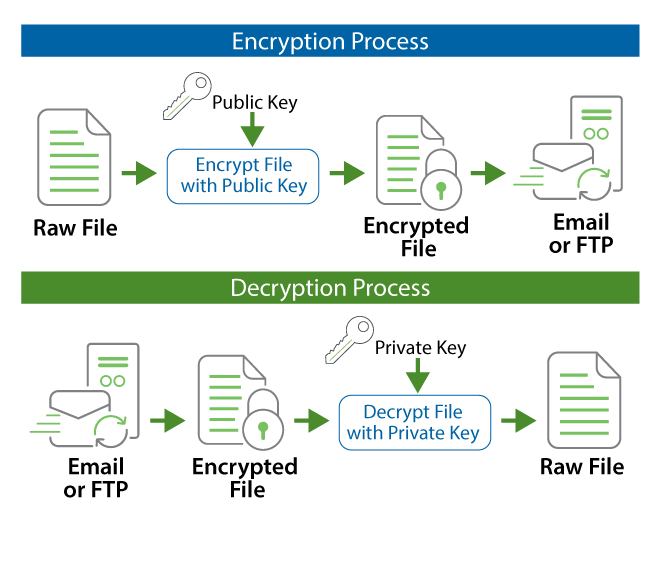
he purpose of this project is to identify the different techniques of data encryption. This project attempts to provide data security. The project consists of creation of code which can secure the data using’ Encryption’. The results matched with the expected output. The project along with the research concluded Data Encryption method as an effective approach towards Data Security.

**INTRODUCTION AND THEORY**

**Encryption is the conversion of data from a readable form, called plaintext, into a form, called ciphertext that cannot be easily understood by unauthorized people.**

**Decryption is the process of converting encrypted data back into its original form (plaintext), so it can be understood.**

* Encryption can protect stored data (on servers, desktops, laptops, tablets, smartphones, portable devices, etc.) and transmitted data (over wired and wireless networks, including the Internet and e-mail).
* Encryption uses a mathematical formula to convert the readable plaintext into unreadable ciphertext. The mathematical formula is an **algorithm** (called a cipher). Decryption is the reverse process that uses the same algorithm to transform the unreadable ciphertext back to readable plaintext.
* The algorithms are built into encryption programs – users don’t have to deal with them when they are using encryption.



* **Encryption keys** are used to implement encryption for a specific user or users. A key generator that works with the selected encryption algorithm is used to generate a unique key or keys for the user(s).
* A key is just a line or set of data that is used with the algorithm to encrypt and decrypt the data.
* Protection is provided by use of the algorithm with the unique key or keys.
* The process is called **secret key**or**symmetric key encryption** where the same key is used with an algorithm to both encrypt and decrypt the data.
* With secret key encryption, it is critical to protect the security of the key because it can be used by anyone with access to it to decrypt the data.
* Where a **key pair** is used, one to encrypt the data and a second one to decrypt the data, the process is called **asymmetric encryption**.
* For this kind of encryption, a key generator is used to generate a unique key pair, one for encryption (a public key) and the other for decryption (a private key).
* With key pairs, it is critical to protect the private decryption key since anyone with access to it can decrypt the data.

**FLOW OF PROGRAM**

* The program starts with the header files being declared.
* On entering the program, the user is given a choice using switch case, as to what the user wants to encrypt. If the user enters 1, then an entered string is encrypted, whereas entering 2 will encrypt an existing file.
* The program has two keys, a public key and a private key. The public key is accepted from the user, while the private key isn’t revealed.
* On accepting the key, the encryption logic begins.
* Our main encryption algorithm consists of a variable storing the multiplication of the square of the ASCII value of each character, and the public key.
* The encrypted value is printed on the screen and stored in an array.
* If the user had chosen to encrypt an existing file, then the contents of the file are updated accordingly.
* class Encrypt ends and class Decrypts begins.
* A for loop does the exact opposite steps as that of the encryption for loop, and the decrypted value is displayed on the screen.
* If the user had chosen to encrypted an existing file, the changes in the file are made accordingly.
* The main() function is used to make an object of the Decrypt class and the methods are called.

**SOURCE CODE**

#include<iostream>

#include<string>

#include<fstream>

#include<cmath>

using namespace std;

/\*This the class used to the encrypt the file or the entered string\*/

class Encrypt

{

public:

int choice,key1,ctr=0,k=0,flag=0;

string s;

unsigned char ctemp;

double number,enc,Log,store[100];

fstream f; //creating an object f of the fstream class

void EncryptData()

{

cout<<"\t\tENCRYPTION\n\n";

cout<<"How would you like the file to be encrypted?\n";

cout<<"1.Entering the string\n";

cout<<"2.Encrypting an existing file\n";

cout<<"\n\nEnter your choice : ";

cin>>choice;

switch(choice)

{

case 1 :

{

cout<<"\nEncrypting an entered string\n";

cout<<"\nPlease enter the string : ";

cin.ignore();

getline(cin,s); //taking the string as input

cout<<"\nThe entered string : "<<s<<endl;

break;

}

case 2 :

{

cout<<"\nEncrypting an existing file\n";

f.open("example.txt"); //opening the existing file

getline(f,s,'\0');

cout<<"\nContents of the file before encrypting : \n";

cout<<s<<endl;

f.close(); //closing the file

flag=1;

break;

}

default : cout<<"Please enter a valid choice!\n";

}

cout<<endl<<"Please enter the key : ";

cin>>key1;

cout<<"The encrypted value : "<<endl<<endl;

for(int i=0;i<s.length();i++)

{

enc=0.0;

ctemp = s[i];

number = (int)ctemp;

enc = ((pow(number,2)\*key1));

Log = log10(enc);

store[i]=Log;

ctr++;

cout<<store[i];

}

if(flag==1)

{

f.open("example.txt");

while(k<ctr)

{

f<<store[k]; //writing the encrypted data to the file

k++;

}

f.close();

}

cout<<endl;

}

};

/\*This the class used to the decrypt the file or the entered string\*/

class Decrypt : public Encrypt //using inheritance

{

public:

int l=0;

double number2,dec;

unsigned char ctemp2;

char store2[100];

void DecryptData()

{

cout<<"\n\n\t\t DECRYPTION \n\n";

cout<<"The Decrypted Value : "<<endl;

for(int j=0;j<ctr;j++)

{

number2 = 0;

number2 = pow(10,store[j]); //taking the antilog

number2 = number2/key1;

dec = pow(number2,(0.5));

ctemp2 = (char)dec;

store2[j]=ctemp2;

cout<<ctemp2<<endl; //displaying the original characters

}

cout<<endl<<endl;

if(flag==1)

{

f.open("example.txt",ios::out|ios::trunc);

while(l<ctr)

{

f<<store2[l]; //writing the decrypted data to the file

l++;

}

f.close();

}

}

};

int main()

{

char ch;

Decrypt obj; //creating an object of the Decrypt class

obj.EncryptData();

cout<<endl<<"Do you want to decrypt the string right now? Y/N : ";

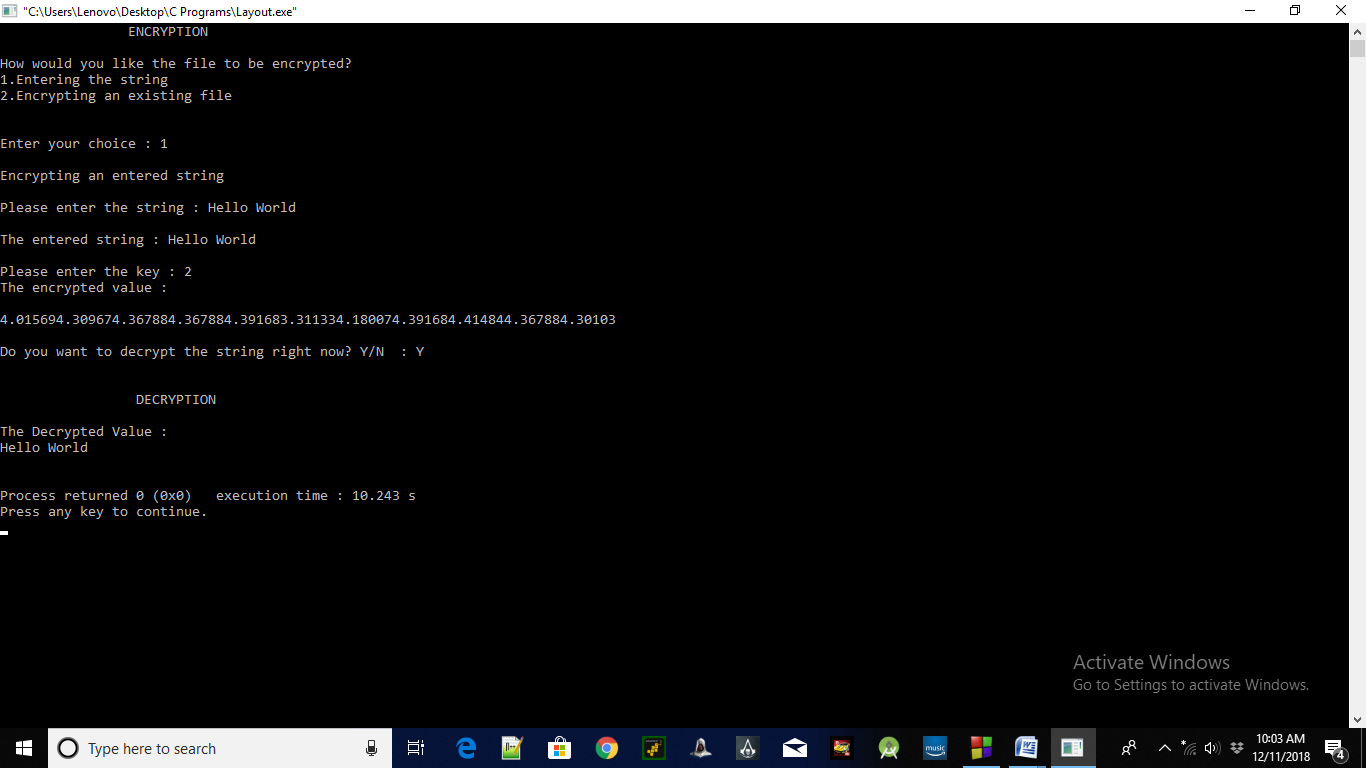
cin>>ch;

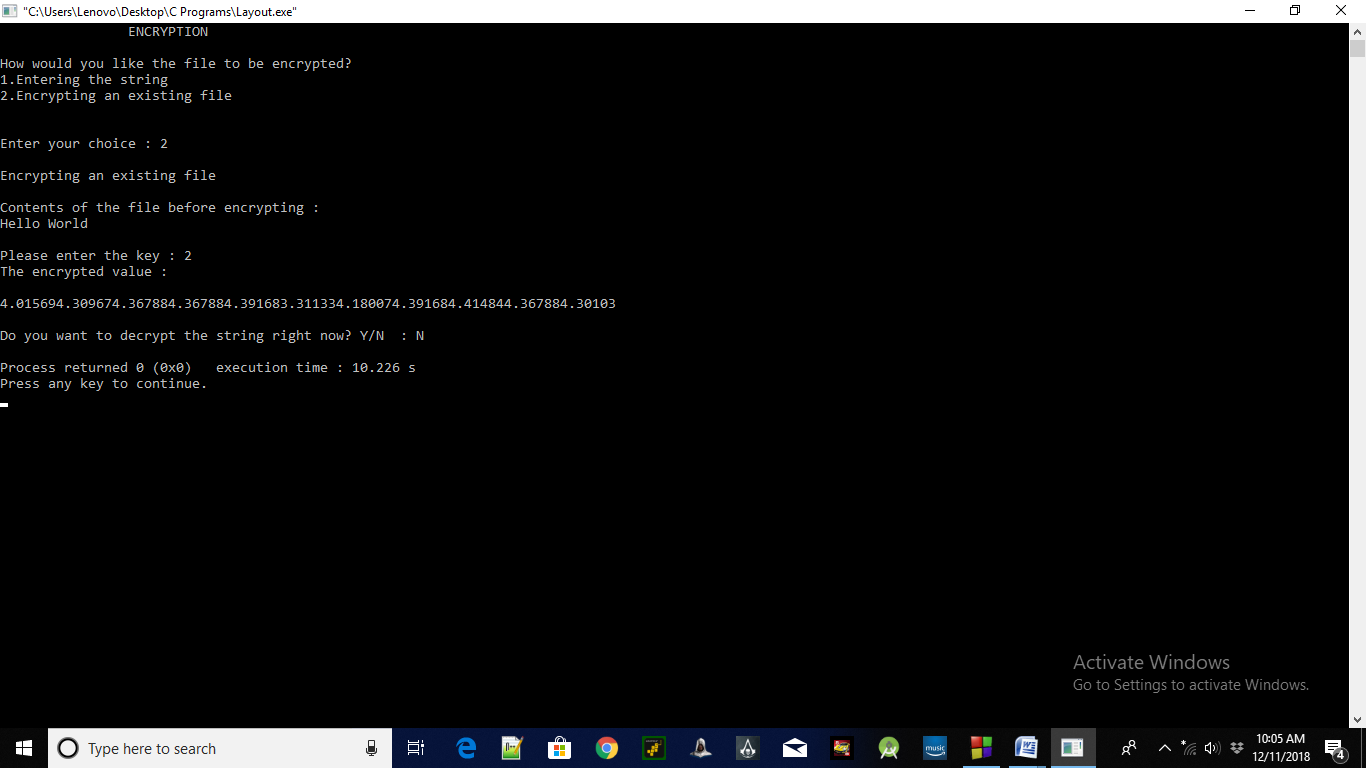
if(ch=='Y'||ch=='y')

obj.DecryptData();

return 0;

}

**OUTPUT**



**LIST OF THE TOPICS / CONCEPTS WHICH ARE COVERED FROM THE SYLLABUS**

* Classes and Objects
* Access Modes / Specifiers
* Functions
* Switch Case
* Strings
* Arrays
* File Handling
* If…else
* For loop
* While loop
* Inheritance

**CONCLUSION**

After looking at the given programs, we can now conclude that encrypting a file is one of the best methods to secure it from unauthorized access by third parties. We have seen how encryption can help secure our data and prevent security breaches.

Even though no method of security is 100% foolproof, encrypting a file is one way in which the file transfers can be made much more secure.

**REFERENCES**

* [www.stackoverflow.com](http://www.stackoverflow.com)
* [www.geeksforgeeks.com](http://www.geeksforgeeks.com)
* [www.tutorialspoint.com](http://www.tutorialspoint.com)
* [www.codesdope.com](http://www.codesdope.com)
* [www.codescracker.com](http://www.codescracker.com)