|  |
| --- |
|  |
|  |
|  |

|  |  |
| --- | --- |
| CS Project  By: Varad and Ketul |  |

# INDEX

|  |  |  |
| --- | --- | --- |
| Sr No | Content | Page No. |
| 1. | Acknowledgement | 4 |
| 2. | Project Specification | 5 |
| 3. | Software Specification | 5 |
| 4. | Code | 6 |
| 5. | Output | 8 |
| 6. | Bibliography | 9 |

# Acknowledgement

I would like to thank the principal Mr. A.P Sharma, for providing us with the

opportunity to develop this brilliant project, which has helped us to attain a deeper understanding of C++.

I would also like to express my special gratitude to the Computer Science

Department in our school and my teacher Mrs. Arshi Habiba for supporting and providing us with all the help required to complete this project.

Finally, I would like to thank my family and friends, valuable suggestions that have helped us complete this project within the limited time frame.

# Project Specification

Privacy is one of the most important features in the current world. Every message you send, can be tracked by the hackers sitting in different parts of the world. Hence, it becomes very important for you to encrypt your message. In simple terms, it means to send the message in such a way, which only you (and the receiver) can understand. Hence, here we have attempted to make one simple Encryption and Decryption Algorithm.

Basically, what it does is it takes two matrices from the user, multiplies it and uses the new matrix formed as a key to encrypt the information.

Similarly, while decrypting the message, it subtracts the same key from the encrypted message and the real message is shown.

Encryption and Decryption find uses in field of Cryptography. Encryption/Decryption is also used by the governments and soldiers to communicate. The reason Encryption/Decryption is famous because:

1. Helps you to protect your confidential data such as passwords and login IDs.
2. Provides confidentiality of private information.
3. Helps you to ensure that that the document or file has not been altered.
4. Encryption process also prevents plagiarism.
5. Helpful for network communication (like the internet) and where a hacker can easily access unencrypted data.

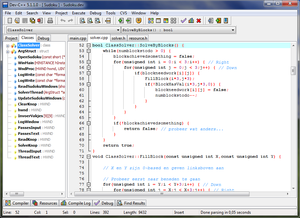
# Software Specification

Here, we have used Dev C++ (version 5.11) to develop our algorithms.

Dev-C++ is a free full-featured integrated development environment (IDE) distributed under the GNU General Public License for programming in C and C++. It is written in Delphi.

Dev C++ is uses, the MinGW or TDM-GCC 64bit port of the GCC as its compiler. Dev-C++ can also be used in combination with Cygwin or any other GCC-based compiler.

Dev-C++ is generally considered a Windows-only program, but there are attempts to create a Linux version: header files and path delimiters are switchable between platforms.



# Code

#include <iostream>

#include <string.h>

#include <cmath>

#include <stdlib.h>

#include <time.h>

#include <process.h>

#include <iomanip>

#include <fstream>

#include <windows.h>

#include <unistd.h>

using namespace std;

class Data

{

char name[100];

int age;

long int salary;

char job[100];

public:

void AcceptData()

{

cout<<"Enter name : ";

fflush(stdin);

gets(name);

cout<<"Enter age : ";

cin>>age;

cout<<"Enter job : ";

fflush(stdin);

gets(job);

cout<<"Enter Salary : ";

cin>>salary;

}

void PrintData()

{

cout<<"Name : ";

puts(name);

cout<<"Age : "<<age<<endl;

cout<<"Job : ";

puts(job);

cout<<"Salary : "<<salary<<endl<<endl;

}

char\* getname()

{

return name;

}

char\* getjob()

{

return job;

}

int getage()

{

return age;

}

int getsalary()

{

return salary;

}

void MatrixEncryption(int Key[2][2]);

void MatrixDecryption(int Key[2][2]);

}D;

int s= (D.getsalary()%10)+1.5;

void Data::MatrixEncryption(int Key[2][2])

{

int i,a;

for(i=0; i<strlen(name);i++)

{

name[i] += Key[i%2][i%2];

}

for(i=0; i<strlen(job);i++)

{

job[i] += Key[i%2][i%2];

}

salary = pow(salary,s);

age = pow(age,s);

age += Key[0][1];

salary += Key[1][0];

}

void Data::MatrixDecryption(int Key[2][2]) // Matrix Decryption Function

{

char choice;

int i,j;

salary-=Key[1][0];

age-=Key[0][1];

salary = pow(salary,1/s);

age = pow(age,1/s);

for(i=0; i<strlen(name);i++)

{

name[i] -= Key[i%2][i%2];

}

for(i=0; i<strlen(job);i++)

{

job[i] -= Key[i%2][i%2];

}

}

void KeyCreation1(int A[2][2])

{

int i,j;

for(i=0;i<2;i++)

{

for(j=0;j<2;j++)

A[i][j]=rand();

}

}

void MatrixEncryption(int C[2][2])

{

int i,j,k,length;

int A[2][2];

int B[2][2];

cout<<"Generating Encryption Key";

sleep(1);

cout<<".";

sleep(1);

cout<<".";

sleep(1);

cout<<"."<<endl<<endl;

sleep(3);

KeyCreation1(A); //FUNCTION TO CREATE THE KEY KeyCreation1(B);

for(i = 0; i < 2; ++i)

for(j = 0; j < 2; ++j)

{

C[i][j]=0;

}

// Multiplying matrix A and B and storing in array C to create the key.

for(i = 0; i < 2; ++i)

for(j = 0; j < 2; ++j)

for(k = 0; k < 2; ++k)

{

C[i][j] += A[i][k] \* B[k][j];

}

D.MatrixEncryption(C); //CALLING THE ENCRYPTION FUNCTION AND ENCRYPTING

D.PrintData();

}

void EncryptionMenu(int C[2][2])

{

int choice;

cout<<"Encryption Menu"<<endl<<endl;

cout<<"1.Matrix Encryption"<<endl;

cout<<"2.Exit"<<endl;

cout<<"Enter choice : "<<endl;

cin>>choice;

switch(choice)

{

case 1: MatrixEncryption(C);

break;

case 2: exit(0);

break;

}

}

void DecryptionMenu(int C[2][2])

{

int choice;

cout<<"Decryption Menu : \n"<<endl; //MENU

cout<<"1.Matrix Decryption"<<endl;

cout<<"2.Exit"<<endl<<endl;

cout<<"Enter choice : ";

cin>>choice;

cout<<endl;

switch(choice)

{

case 1: D.MatrixDecryption(C);

D.PrintData();

break;

case 2: exit(0);

break;

}

}

int main()

{

D.AcceptData();

int choice;

int C[2][2];

do{

cout<<"Menu : \n"<<endl; //MENU

cout<<"1.Encryption"<<endl;

cout<<"2.Decryption"<<endl;

cout<<"3.Exit"<<endl<<endl;

cout<<"Enter choice : ";

cin>>choice;

cout<<endl;

switch(choice)

{

case 1: EncryptionMenu(C);

break;

case 2: DecryptionMenu(C);

break;

case 3: break;

}

}while(choice!=3);

return 0;

}

# Output

# 

# Bibliography

# Sumita Arora C++ Class 11 and 12

# E. Balagurusamy C++