/\*1. Vigenere Cipher

Name: Akshay Brahmankar Date: 29-01-16

Roll No: 12(BE-H) Batch-H1 Sub: Network Security(LAB) \*/

#include <iostream>

#include<string>

using namespace std;

int main()

{

string str1,key1;

cout << "vigenere cipher" << endl;

int col,row;

col = row = 62;

char temp[] = "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789";

char mat[col][row];

for(int i=0;i<col;i++){

for(int j=0;j<row;j++){

mat[i][j]= 0;

}

}

int key =0,key2=0;

int cnt=0;

for(int i=0;i<col;i++){

key=i;

for(int j=0;j<row;j++){

mat[i][j] = temp[key%62];

key++;

}

}

for(int i=0;i<col;i++){

for(int j=0;j<row;j++){

cout << " " << mat[i][j];

}

cout<<" "<<endl;

}

cout<<endl<<endl;

cout << "Enter alphanumeric plaintext: " << endl;

cin >> str1;

cout << "Enter the key: "<<endl;

cin >> key1;

cout << "Encrypted text is: "<<endl;

int k=0,i=0;

string encrypt\_result="";

string decrypt\_result="";

int p,q;

while( str1[i]!= 0){

//for selecting column

if(str1[i]>='A' && str1[i]<='Z'){

q = (str1[i]-'A')+1;

}

else if(str1[i]>='a' && str1[i]<='z'){

q = (str1[i]-'a')+1+26;

}

else if(str1[i]>='0' && str1[i]<='9'){

q = (str1[i]-'0')+1+52;

}

//for selecting row

if(key1[k%(key1.length())]>='A' && key1[k%(key1.length())]<='Z'){

p = key1[k%(key1.length())]-'A';

}

else if(key1[k%(key1.length())]>='a' && key1[k%(key1.length())]<='z'){

p = (key1[k%(key1.length())]-'a');

}

else if(key1[k%(key1.length())]>='0' && key1[k%(key1.length())]<='9'){

p = (key1[k%(key1.length())]-'0');

}

encrypt\_result+=mat[p][q];

i++;

k++;

}

cout<< encrypt\_result<<endl;

//decryption ..............!!

cout<< "Decryption result is: "<<endl;

i=0;k=0;

while(encrypt\_result[i]!=0){

if(encrypt\_result[i]>='A' && encrypt\_result[i]<='Z'){

q = (encrypt\_result[i]-'A')+1;

}

else if(encrypt\_result[i]>='a' && encrypt\_result[i]<='z'){

q = (encrypt\_result[i]-'a')+1-26-2;

}

else if(encrypt\_result[i]>='0' && encrypt\_result[i]<='9'){

q = (encrypt\_result[i]-'0')+1-52-2;

}

////////////////

if(encrypt\_result[i]>='A' && encrypt\_result[i]<='Z'){

p = 60-(key1[k%(key1.length())]-'A');

}

else if(encrypt\_result[i]>='a' && encrypt\_result[i]<='z'){

p = 60-(key1[k%(key1.length())]-'a');

}

else if(encrypt\_result[i]>='0' && encrypt\_result[i]<='9'){

p = 60 -(key1[k%(key1.length())]-'0');

}

i++;

k++;

decrypt\_result+=mat[p][q];

}

cout<< decrypt\_result <<endl;

return 0;

}

/\*Output:

Enter alphanumeric plaintext:

ABC

Enter the key:

C

Encrypted text is:

DEF

Decryption result is:

ABC

Enter alphanumeric plaintext:

AB123

Enter the key:

CT

Encrypted text is:

DV4M6

Decryption result is:

AB123

Process returned 0 (0x0) execution time : 7.460 s

Press any key to continue.

Process returned 0 (0x0) execution time : 5.856 s

Press any key to continue.

\*/