МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

НАВЧАЛЬНО-НАУКОВИЙ КОМПЛЕКС

«ІНСТИТУТ ПРИКЛАДНОГО СИСТЕМНОГО АНАЛІЗУ»

Лабораторна робота № 1

з курсу «Технології захисту інформації»

Тема: «Дослідження шифрів підстановки»

Виконав:

студент IV курсу

групи ДА-32

Колінько Анжела

Київ – 2017

Варіант 11

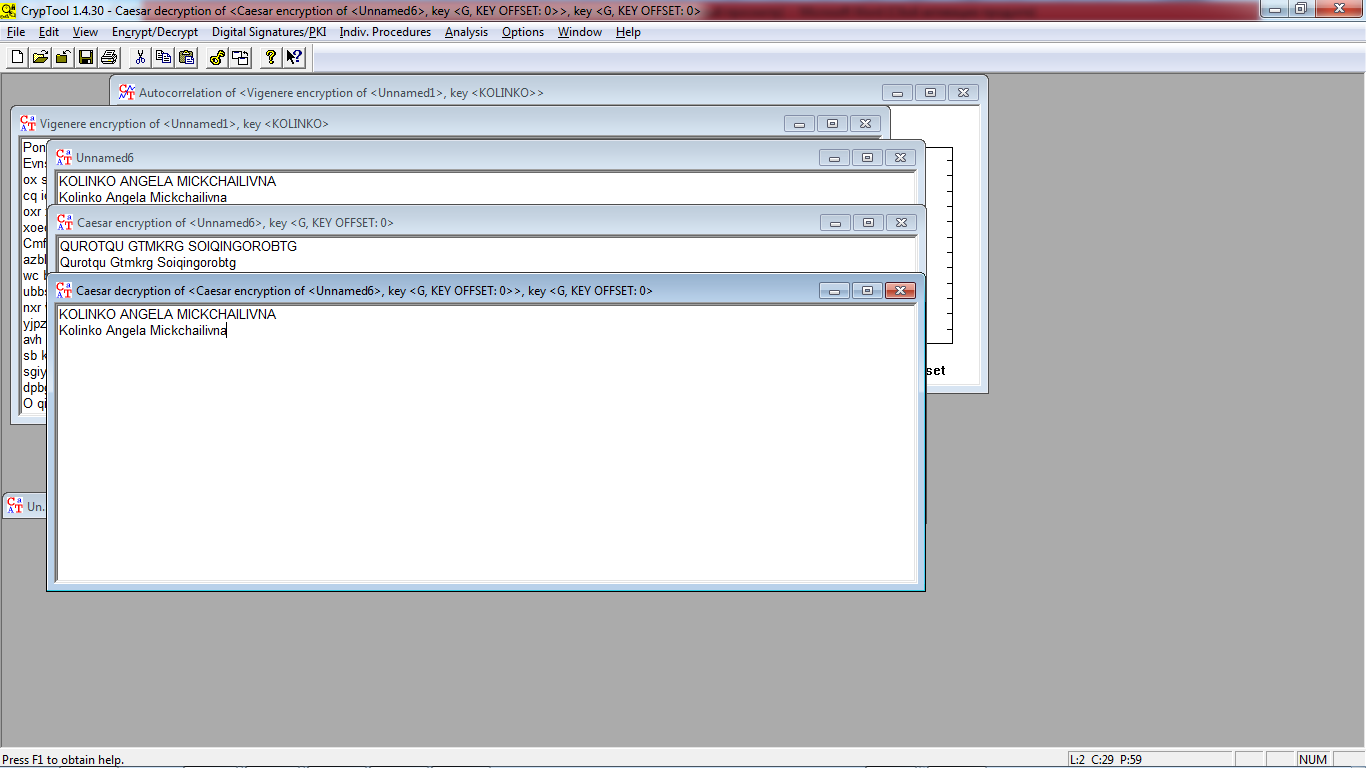
Мета: дослідження шифрів підстановки та методів їх криптоаналізу з використанням програмних засобів CrypTool і їх практична реалізація.

1. Результати ручного шифрування і дешифрування ПІБ шифром Цезаря.

Ключ K обчислюється за формулою K = (N + n + c) mod 26, де N - номер групи, n - номер студента за списком, c = 15 - константа. K = (N + n + c) mod 26=(32+11+15) mod 26=58 mod 26=6;

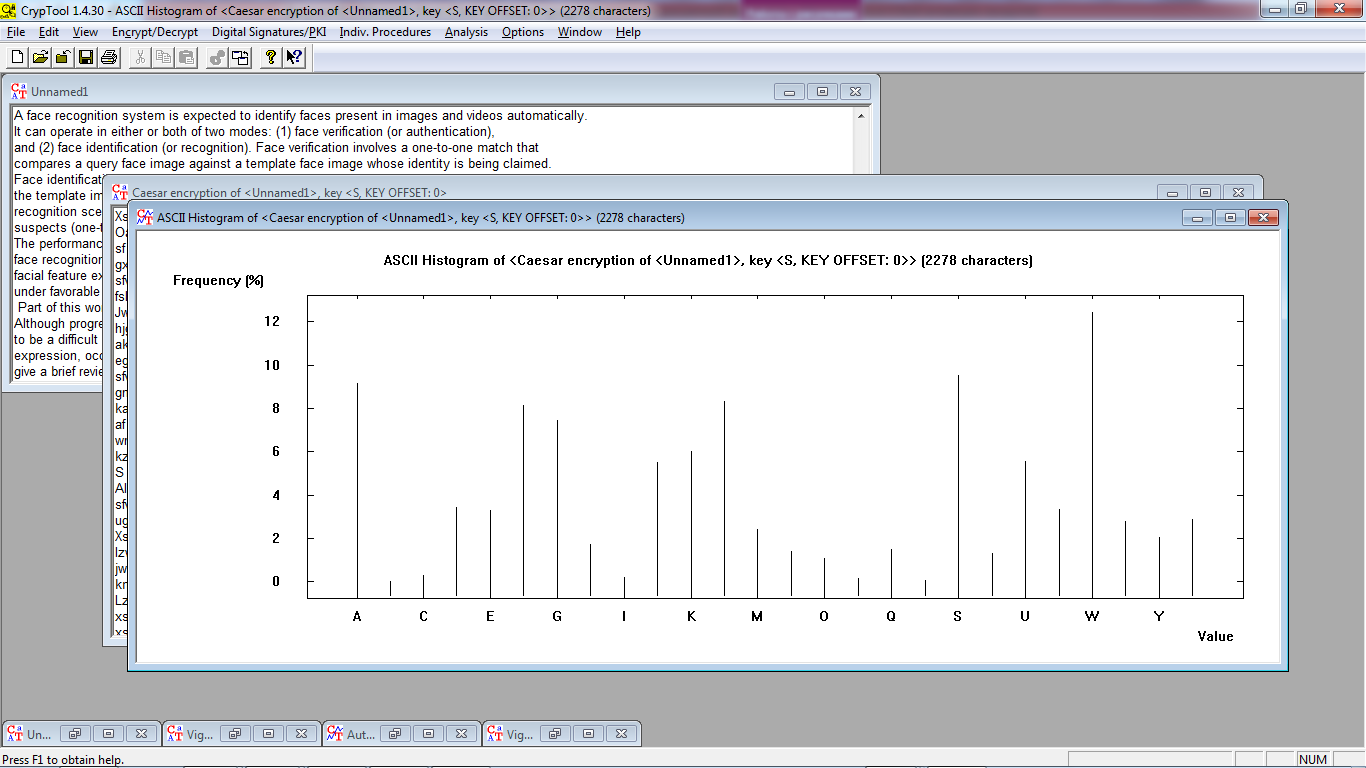
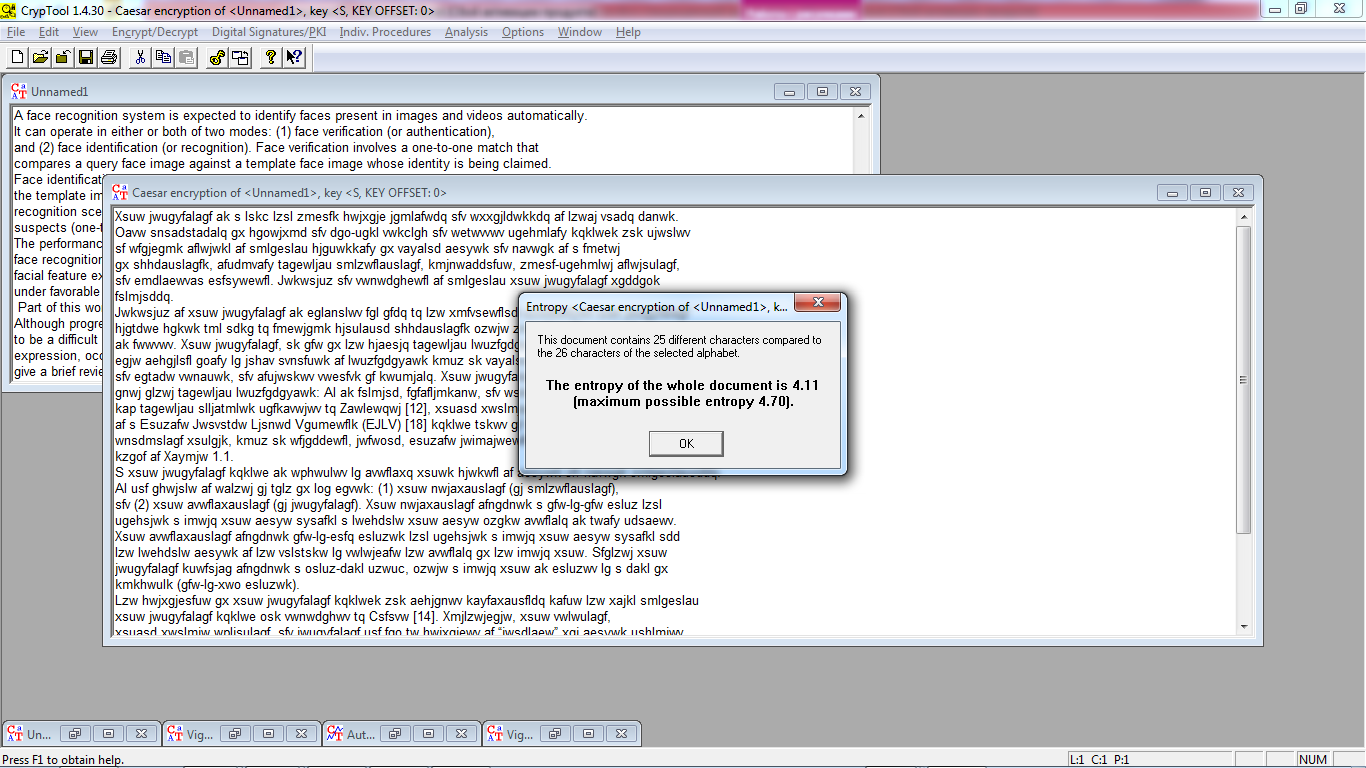
KOLINKO ANGELA MICKCHAILIVNA -> QUROTQU GTMKRG SOIQINGOROBTG -> KOLINKO ANGELA MICKCHAILIVNA

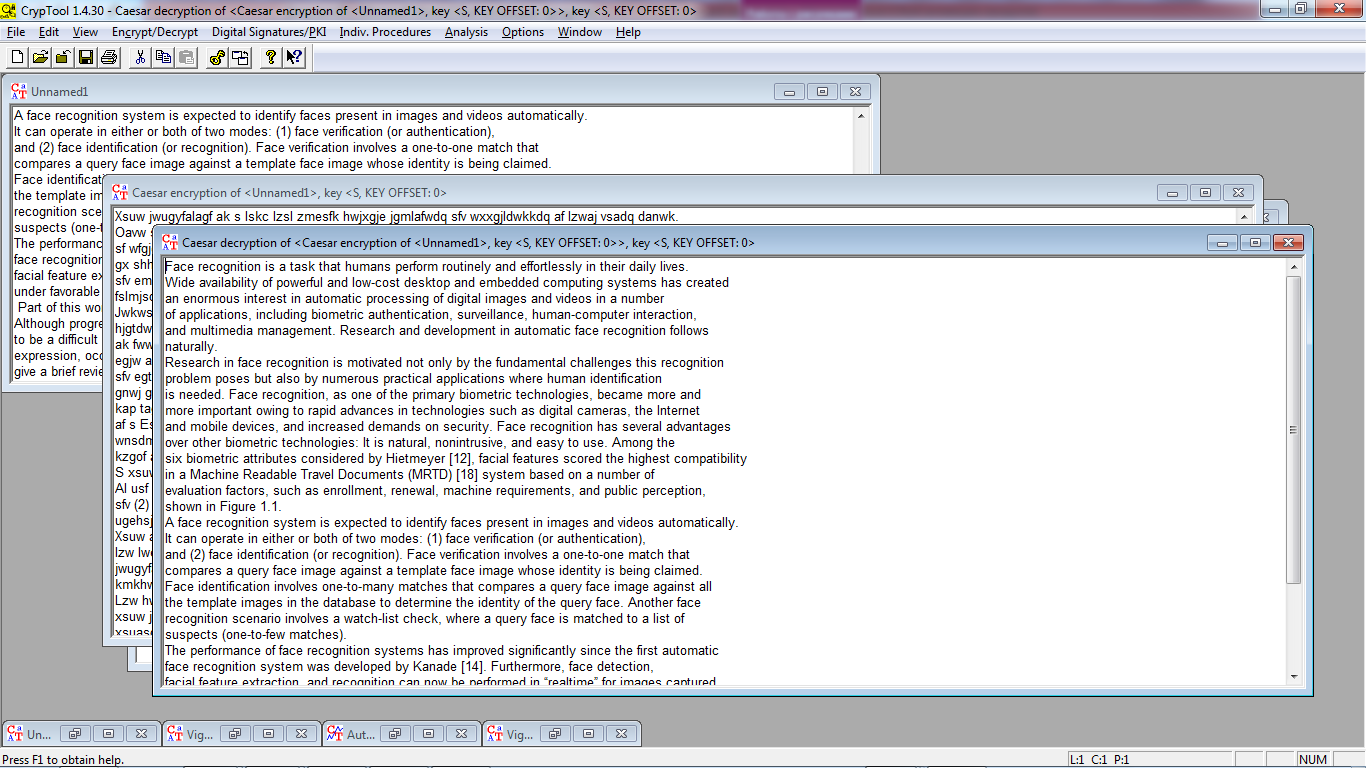
2. Результати шифрування і дешифрування ПІБ шифром Цезаря з використанням пакету CrypTool.



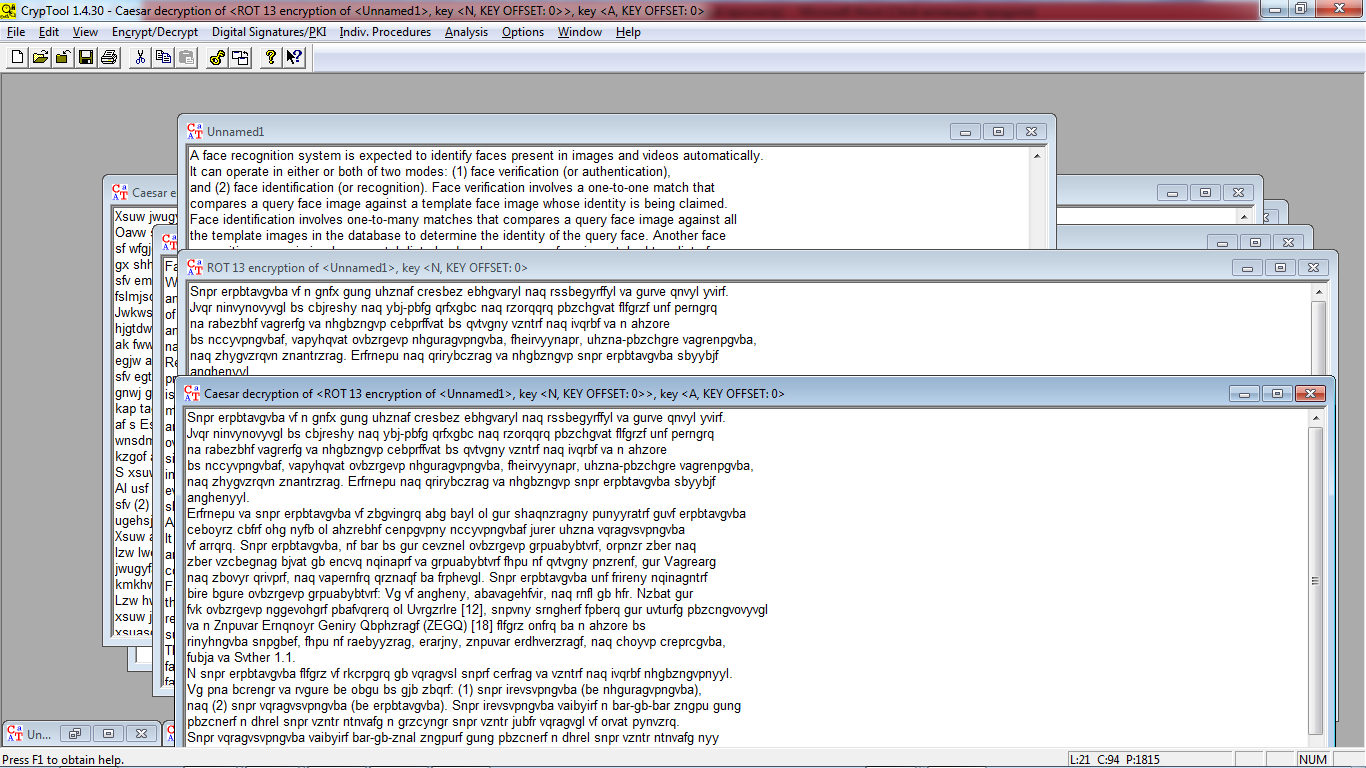
3. Результати шифрування і дешифрування текстового файлу шифром Цезаря з використанням пакету CrypTool, ентропія і максимально можлива ентропія, гістограма розподілу частоти символів.

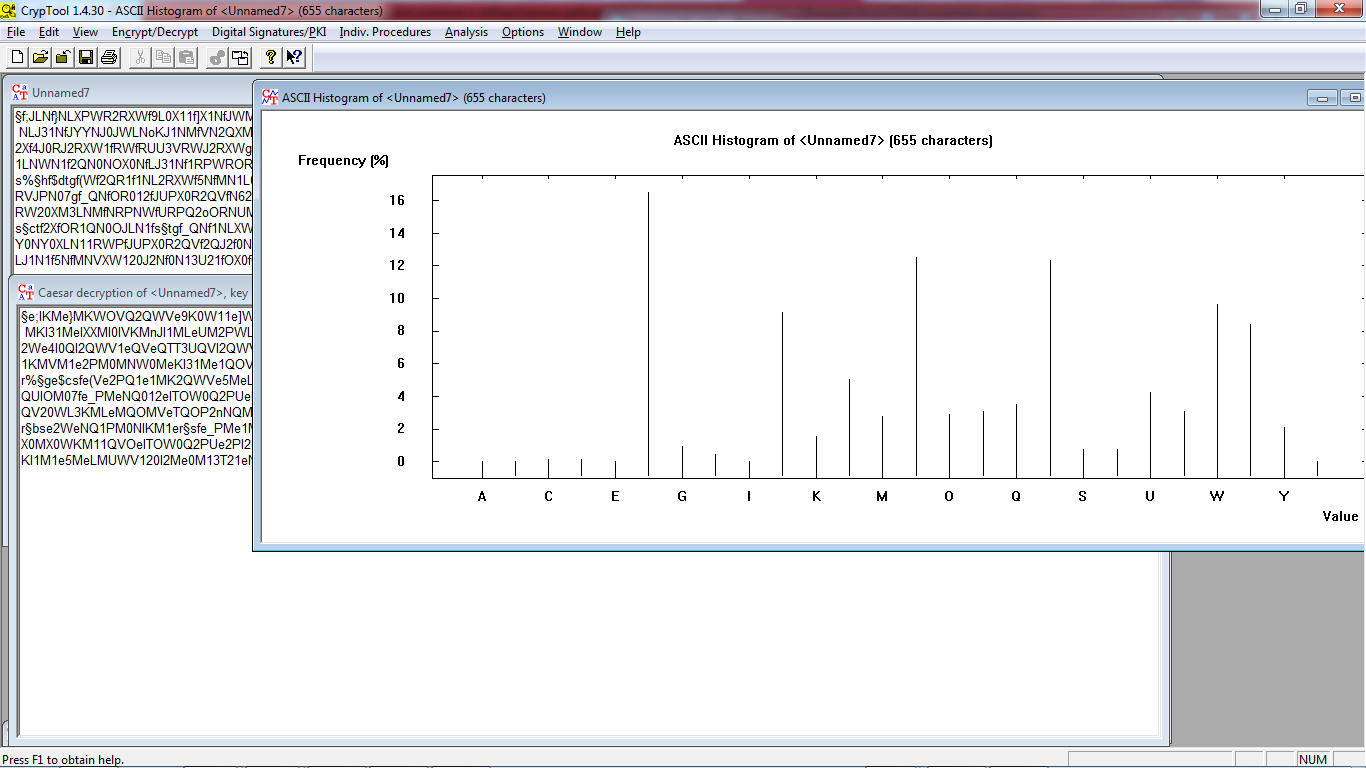
Файл з текстом обирається за номером варіанту, а ключ шифрування - (№ групи + № списку + день народження) % 26=(32+11+1) % 26=44%26=18



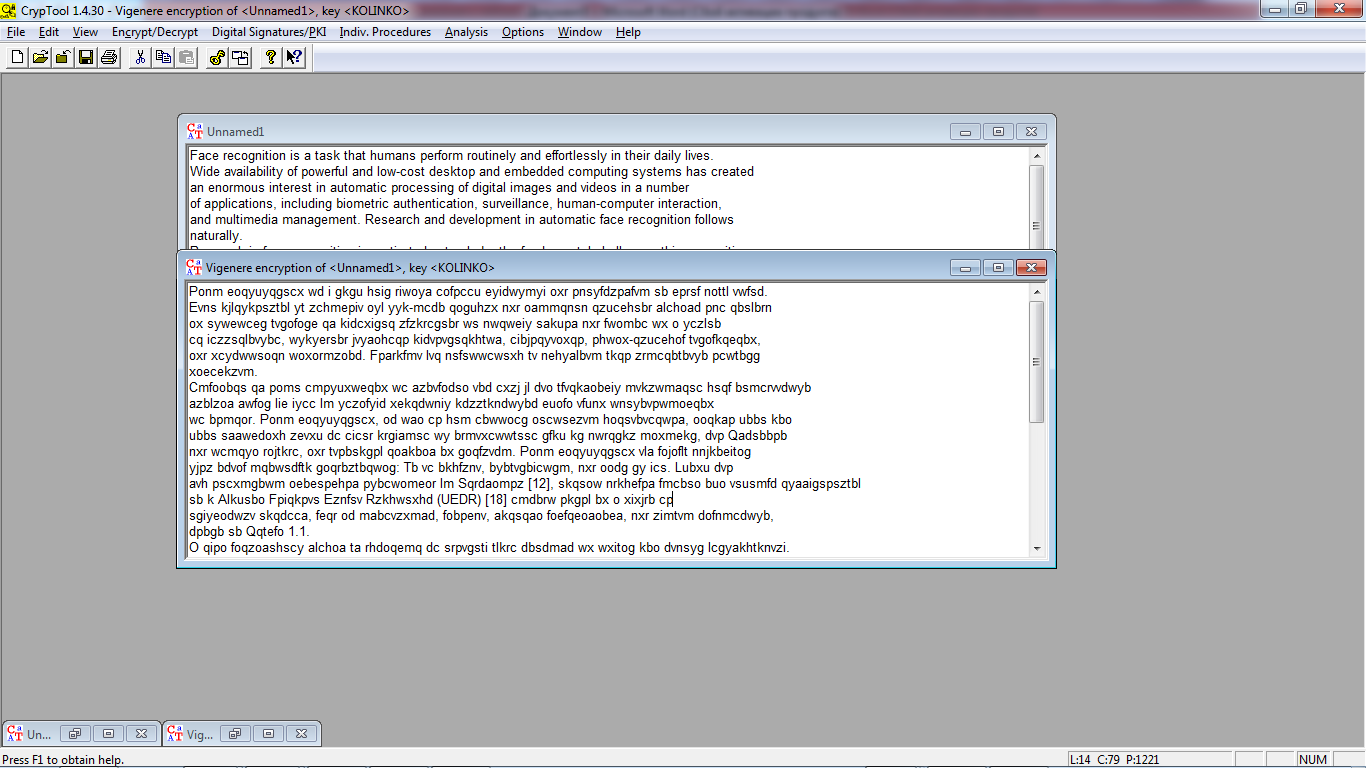


4. Результати шифрування і дешифрування текстового файлу шифром Rot-13 з використанням пакета CrypTool.

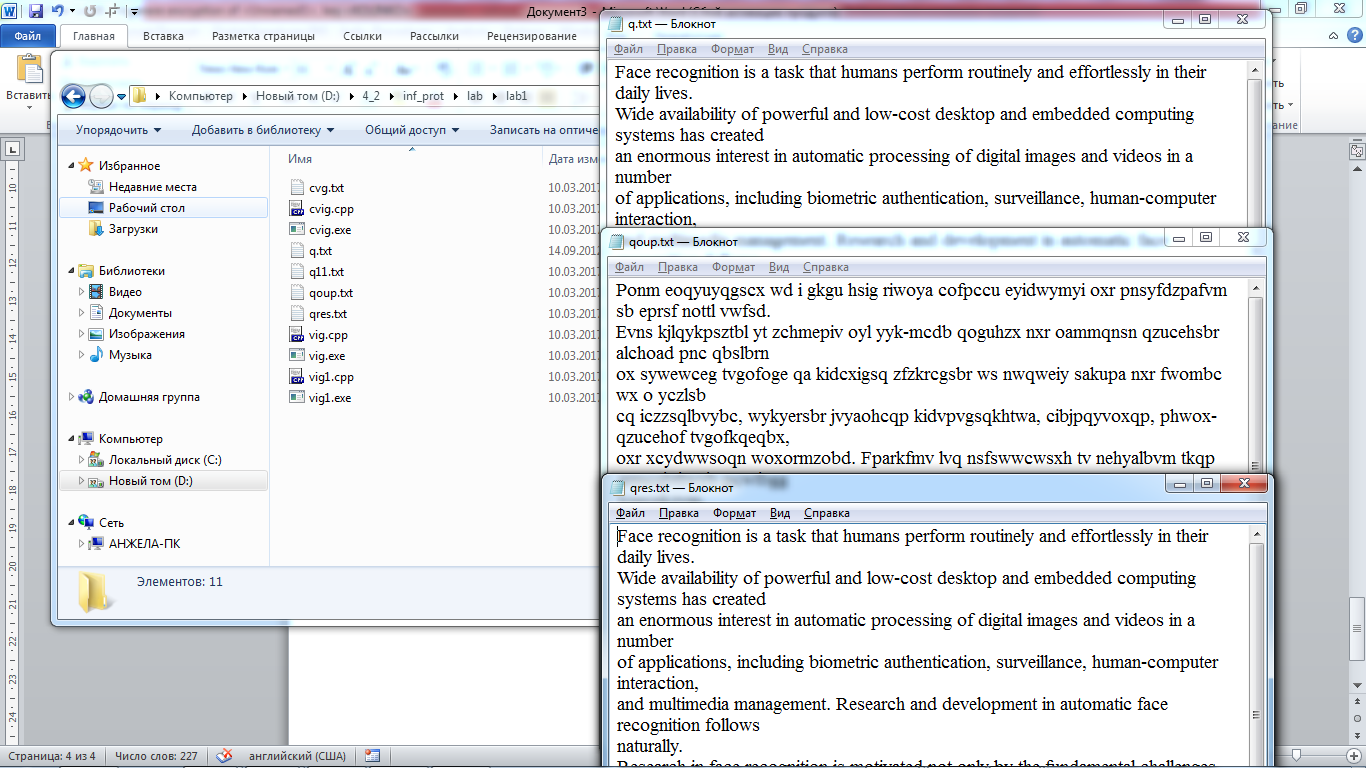




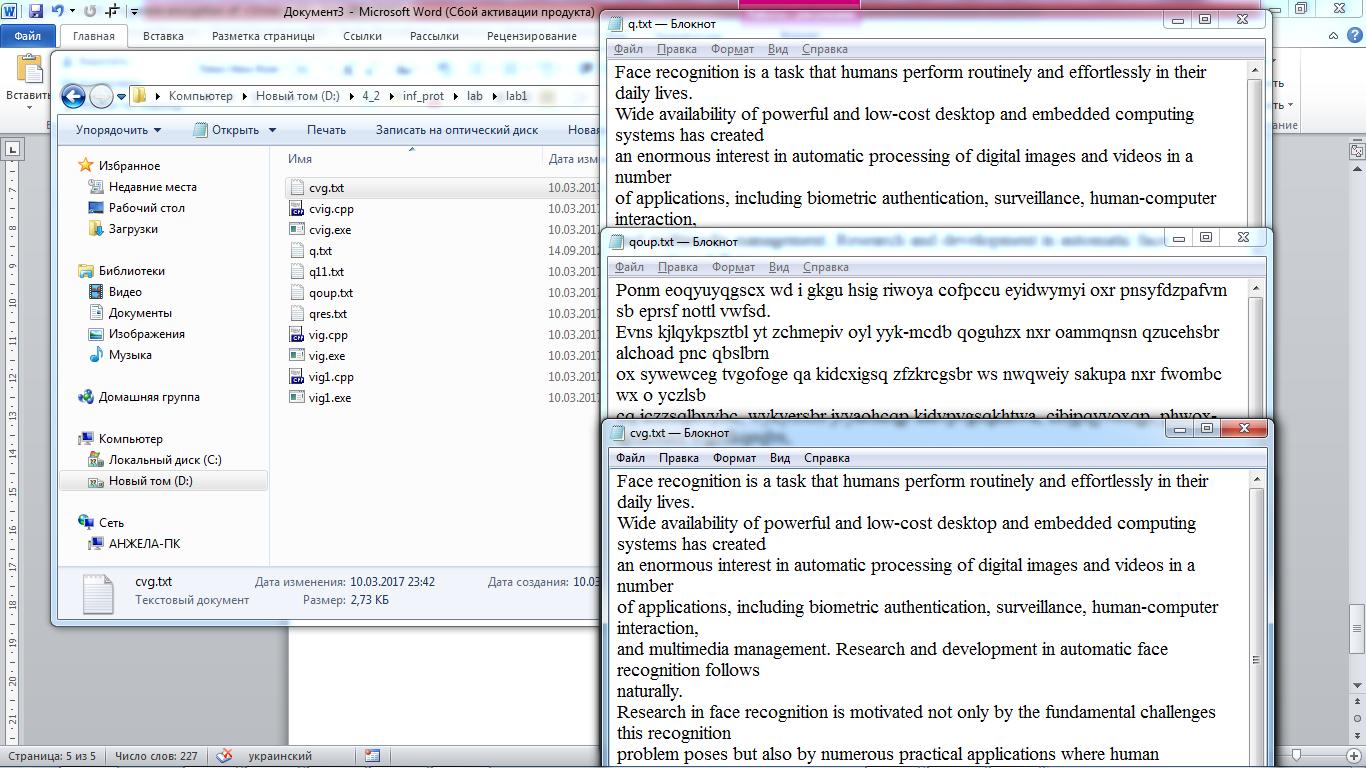
5. Результати шифрування і дешифрування текстового файлу шифром Віженера з використанням пакета CrypTool.



6. Результати шифрування і дешифрування текстового файлу шифром Віженера з використанням власної програми.



7. Результати дешифрування текстового файлу атакою Беббіджа.



Висновок: на лабораторній роботі були розглянуті шифри Цезаря та Віженера.

Лістинг 1. Шифрування та дешифрування шифром Віженера.

#include<iostream>

#include<string>

#include<fstream>

#define key "kolinko"

#define engal 26

using namespace std;

typedef unsigned short int usi;

bool cvig(string& ifn,string& dst/\*,string& key\*/){

ifstream fpi; ofstream fpo;

fpi.open(ifn.c\_str(),ios::binary);

if(!fpi.is\_open()) {cout<<"\nerr: gff no input file"; return 0;}

else{

fpo.open(dst.c\_str(),ios::binary);

if(!fpo.is\_open()) {cout<<"\nerr: gff no input file"; return 0;}

else{

unsigned char ch='\0'; unsigned long ulcnt=0; bool uc=0;

while(1){

fpi>>noskipws>>ch;

if(fpi.eof()) break;

if(((usi)ch<'A')||((usi)ch>'z')||((usi)ch>'Z'&&(usi)ch<'a'))//skip letters that is not in alphabet;

fpo<<ch;

else{

if(isupper(ch)){uc=1; ch=tolower(ch);}

else uc=0;

unsigned char tmpch=(ch+key[ulcnt%(sizeof(key)-1)]-2\*'a')%engal+'a';

/\*cout<<"\nch+key[ulcnt%(sizeof(key)-1)]-2\*'a')%engal+'a'="<<

(unsigned char)ch<<'+'<<key[ulcnt%(sizeof(key)-1)]<<"-2\*a%"<<engal<<"+a=\n"<<

(usi)ch<<'+'<<(usi)key[ulcnt%(sizeof(key)-1)]<<"-2\*"<<(usi)'a'<<'%'<<engal<<'+'<<(usi)'a'<<"=\n"<<

(usi)((ch+key[ulcnt%(sizeof(key)-1)]-2\*'a')%engal+'a')<<'='<<

(unsigned char)((ch+key[ulcnt%(sizeof(key)-1)]-2\*'a')%engal+'a');\*/

if(uc) tmpch=toupper(tmpch);

fpo<<tmpch;

ulcnt++;

}

}

fpi.close(); fpo.close();

}

}

return 1;

}

bool decvig(string& ifn,string& dst/\*,string& key\*/){

ifstream fpi; ofstream fpo;

fpi.open(ifn.c\_str(),ios::binary);

if(!fpi.is\_open()) {cout<<"\nerr: gff no input file"; return 0;}

else{

fpo.open(dst.c\_str(),ios::binary);

if(!fpo.is\_open()) {cout<<"\nerr: gff no input file"; return 0;}

else{

unsigned char ch='\0'; unsigned long ulcnt=0; bool uc=0;

while(1){

fpi>>noskipws>>ch;

if(fpi.eof()) break;

if(((usi)ch<'A')||((usi)ch>'z')||((usi)ch>'Z'&&(usi)ch<'a'))//skip letters that is not in alphabet;

fpo<<ch;

else{

if(isupper(ch)){uc=1; ch=tolower(ch);}

else uc=0;

unsigned char tmpch=(ch-key[ulcnt%(sizeof(key)-1)]+engal)%engal+'a';

/\*cout<<"\nch+key[ulcnt%(sizeof(key)-1)]-2\*'a')%engal+'a'="<<

(unsigned char)ch<<'+'<<key[ulcnt%(sizeof(key)-1)]<<"-2\*a%"<<engal<<"+a=\n"<<

(usi)ch<<'+'<<(usi)key[ulcnt%(sizeof(key)-1)]<<"-2\*"<<(usi)'a'<<'%'<<engal<<'+'<<(usi)'a'<<"=\n"<<

(usi)((ch+key[ulcnt%(sizeof(key)-1)]-2\*'a')%engal+'a')<<'='<<

(unsigned char)((ch+key[ulcnt%(sizeof(key)-1)]-2\*'a')%engal+'a');\*/

if(uc) tmpch=toupper(tmpch);

fpo<<tmpch;

ulcnt++;

}

}

fpi.close(); fpo.close();

}

}

return 1;

}

int main(){

string srs="q.txt",dst="qoup.txt",dcdf="qres.txt";

//string key; cout<<"enter key\n"; cin>>key;

cout<<"\nkey="<<key;//<<"\nfdt="<<fdt;

cvig(srs,dst/\*,key\*/);

decvig(dst,dcdf/\*,key\*/);

return 0;

}

Лістинг 2. Атака Беббіджа.

#include<iostream>

#include<conio.h>

#include<string>

#include<fstream>

#define engal 26

using namespace std;

typedef unsigned short int usi;

int gcd ( int a, int b ){

int c;

while ( a != 0 ) {

c = a; a = b%a; b = c;

}

return b;

}

bool gff(string& ifn,string& fdt){

ifstream fpi;

fpi.open(ifn.c\_str(),ios::binary);

if(!fpi.is\_open()) {cout<<"\nerr: gff no input file"; return 0;}

else{

unsigned char ch='\0';

while(1){

fpi>>noskipws>>ch;

if(fpi.eof()) break;

if(((usi)ch<'A')||((usi)ch>'z')||((usi)ch>'Z'&&(usi)ch<'a')) continue;

else fdt+=ch;

}

fpi.close();

}

return 1;

}

bool decvig(string& ifn,string& dst,string& key){

ifstream fpi; ofstream fpo;

fpi.open(ifn.c\_str(),ios::binary);

if(!fpi.is\_open()) {cout<<"\nerr: gff no input file"; return 0;}

else{

fpo.open(dst.c\_str(),ios::binary);

if(!fpo.is\_open()) {cout<<"\nerr: gff no input file"; return 0;}

else{

unsigned char ch='\0'; unsigned long ulcnt=0; bool uc=0;

while(1){

fpi>>noskipws>>ch;

if(fpi.eof()) break;

if(((usi)ch<'A')||((usi)ch>'z')||((usi)ch>'Z'&&(usi)ch<'a'))//skip letters that is not in alphabet;

fpo<<ch;

else{

if(isupper(ch)){uc=1; ch=tolower(ch);}

else uc=0;

unsigned char tmpch=(ch-key[ulcnt%(sizeof(key)-1)]+engal)%engal+'a';

if(uc) tmpch=toupper(tmpch);

fpo<<tmpch;

ulcnt++;

}

}

fpi.close(); fpo.close();

}

}

return 1;

}

int main(){

string srs="qoup.txt",dst="cvg.txt",fdt="",key=""; gff(srs,fdt);

unsigned long int ulcnt=1, ulmchs=0, ulmaxm=0, klarr[5]; usi cntkla=0,kl=0;

for(ulcnt=1;ulcnt<fdt.size();ulcnt++){

//cout<<"\nulcnt="<<ulcnt<<'\n';

for(unsigned long i=0;i<fdt.size()-2;i++){

if(fdt[(ulcnt+i)%fdt.size()]==fdt[i]&&fdt[(ulcnt+i+1)%fdt.size()]==fdt[i+1]&&fdt[(ulcnt+i+2)%fdt.size()]==fdt[i+2]){

ulmchs++;

}

}

if(ulmchs>ulmaxm){

cout<<"\nulcnt="<<ulcnt<<" ulmchs="<<ulmchs<<" ulmaxm="<<ulmaxm<<'\n';// getch();

ulmaxm=ulmchs;

if(ulcnt>2&&cntkla<4){

klarr[cntkla]=ulcnt; cntkla++;

}

}

else{

/\*cout<<"ulmchs="<<ulmchs<<'\n';//<<" ulmaxm="<<ulmaxm<<'\n';

if((ulcnt-1)%7==0) getch();\*/

}

ulmchs=0;

}

cout<<"\nselect key length from ";

for(usi i=0;i<cntkla;i++) cout<<klarr[i]<<' ';

/\*for(usi i=1;i<cntkla;i++){

cout<<"\ngcd="<<gcd(klarr[i],klarr[i-1]);

}\*/

kl=gcd(klarr[1],klarr[0]);

cout<<"\nkl="<<kl;

for(usi i=0;i<kl;i++){

unsigned long mgcnt=0; unsigned char mgch;

for(unsigned long j=i;j<fdt.size();j+=kl){

unsigned long gcnt=0;

for(unsigned long k=j+kl;k<fdt.size();k+=kl){//get most frequently occurring char in group;

if(fdt[j]==fdt[k]) gcnt++;

if(mgcnt<gcnt){

mgcnt=gcnt; mgch=fdt[k];

}

}

}

if(i!=4) {

cout<<"\nmgch="<<(usi)mgch<<" "<<(unsigned char)((mgch-'e'+engal)%engal+'a');

key+=(unsigned char)((mgch-'e'+engal)%engal+'a');

}

else{

cout<<"\nhere is \'i\' most frequently occurring char:\nmgch="<<(usi)mgch<<" "<<(unsigned char)((mgch-'i'+engal)%engal+'a');

key+=(unsigned char)((mgch-'i'+engal)%engal+'a');

}

}

cout<<"\nkey="<<key;

decvig(srs,dst,key);

return 0;

}

/\*unsigned long int ulcnt=0, ulmchs=0, ulmaxm=0,mdst=0;

for(ulcnt=0;ulcnt<fdt.size();ulcnt++){

cout<<"\nulcnt="<<ulcnt<<'\n';

for(unsigned long i=0;i<fdt.size()-2;i++){

if(fdt[ulcnt]==fdt[i]&&fdt[ulcnt+1]==fdt[i+1]&&fdt[ulcnt+2]==fdt[i+2]){

ulmchs++;

cout<<"\nmdst="<<mdst; mdst=i-mdst; cout<<"\tmdst="<<mdst;

if(mdst%7==0) cout<<"\t here db7\n";

}

}

if(ulmchs>ulmaxm){

cout<<"\nulcnt="<<ulcnt<<" ulmchs="<<ulmchs<<" ulmaxm="<<ulmaxm<<"\n\n";

getch();

ulmaxm=ulmchs;

}

else{

cout<<"ulmchs="<<ulmchs<<'\n';//<<" ulmaxm="<<ulmaxm<<'\n';

if((ulcnt-1)%7==0) getch();

}

ulmchs=0; mdst=0;

}\*/