МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ОПТИКО-МЕХАНІЧНИЙ КОЛЛЕДЖ КИЇВСЬКОГО НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ ІМЕНІ ТАРАСА ШЕВЧЕНКА

**ЗВІТ З ЛАБОРАТОРНОЇ РОБОТИ №1**

З НАВЧАЛЬНОЇ ДИСЦИПЛІНИ «ЗАХИСТ ПРОГРАМ ТА ДАНИХ»

**НА ТЕМУ: «БАЗОВІ ШИФРИ. ЧАСТОТНИЙ КРИПТОАНАЛІЗ»**

Виконав студент: Яремко Данііл Михайлович

Група ІПЗ - 31

Спеціальність Інженерія програмного забезпечення

Перевірив викладач: Пугачов О. П.

КИЇВ – 2020

Мета роботи: ознайомитися з базовими шифрами. Розглянути методику частотного криптоаналізу.

Контрольні питання

1. Опишіть шифр Полібія.

Одна з відомих модифікацій шифру простої заміни – квадрат Полібія. Візьмемо алфавіт з 32 букв. Виберемо ключ – будь-яке слово, в якому немає однакових букв. Запишемо його в перші клітинки квадрата розміром, наприклад, 4×8. В останні клітинки запишемо алфавіт за війнятком тих букв, що зустрічаються в ключі. Для зашифрування букви повідомлення замінюються на букви, що стоять під ними в квадраті Полібія.

2. Опишіть шифр простої заміни.

Подальший розвиток шифру Цезаря є очевидним: нижня строчка може бути записана з випадковим порядком букв. Такий шифр носить назву шифру простої заміни. Ключем такого шифру є порядок розташування букв в нижній строчці, так звана «таблиця заміни». Якщо в шифрі Цезаря існує тільки 33 варіанта ключів, то в шифрі простої заміни їх вже 33! (33 факторіал).

3. Опишіть шифр Тритемія.

В XV столітті абат Тритемій (Германія) запропонував шифр на основі “таблиці Тритемія”. Тут перша строчка є одночасно і строчкою букв відкритого тексту. Перша буква тексту шифрується по першій строчці, друга – по другій і т.д. Після останньої строчки знову повертаємось до першої. Наприклад, слово “КРИПТОГРАФИЯ” зміниться на “КСКТЦУИЧЗЭТЙ”. Однак, в такому варіанті, в шифрі Тритемія був відсутній ключ. В подальшому удосконалення шифру пішли по двом шляхам: - введення випадкового порядку розташування букв алфавіту; - застосування більш склад

4. Опишіть шифр перестановки.

Оберемо ціле додатне число, наприклад, 5. Створимо випадкову підстановку: (1 2 3 4 5) (3 2 5 1 4). Зашифруємо фразу: «БАЖАЄШ БАГАТО ЗНАТИ, ТРЕБА МЕНШЕ СПАТИ». Для цього доповнимо фразу до довжини кратної 5 випадковими символами та розіб’ємо на групи по 5 букв. БАЖАЄ ШБАГА ТОЗНА ТИТРЕ БАМЕН ШЕСПА ТИВСЕ Букви кожної групи переставимо згідно обраної підстановки. Отриманий текст запишемо без пропусків. 7 «ААБЄЖГБШААНОТАЗРИТЕТЕАБНМПЕШАССИТЕВ» При розшифруванні текст розбивається на групи по 5 букв і букви переставляються в зворотному порядку. Ключом шифру є степінь підстановки (тут 5) і порядок розташування чисел в нижньому рядку підстановки.

5. Чи є шифр Полібія шифром простої заміни?

Так, він є однією з відомих модифікацій шифру простої заміни.

6. Як залежить стійкість шифру від довжини ключа?

Чим довше і складніше ключ, тим шифр стійкіший.

7. Опишіть метод частотного криптоанализу.

Частотний криптоаналіз базується на застосуванні статистики для аналізу текстової інформації. Текст складається із слів, слова із літер. Кількість літер в кожній мові обмежена. Важливими характеристиками тексту є повторюваність літер, пар літер (біграм) і , взагалі, m-грам, поєднання літер друг с другом, чередування голосних і приголосних і т.д. Всі ці характеристики є достатньо стійкими і можуть бути використані для аналізу шифртекстів.

Процес криптоанализу можна представити наступним чином. Криптоаналітик підраховує частоти букв в шифротексті. Далі він бере в шифртексті символ, що зустрічається найбільш часто, и припускає, що це пробіл. Потім бере наступний символ, що зустрічається найбільш часто, и припускає, що це Е (для англійскої мови), и т.д. Шляхом проб і помилок такий метод може привести до рішення задачі. Крім того, при підставленні букв замість символів аналізованого шифртексту криптоаналітик враховує частоти появи сполучень із двох букв (діаграм), трьох букв (триграмм) і т.д.

8. В яких випадках можна застосовувати метод частотного криптоанализу?

Коли треба розшифрувати текст, букви якого стоять у вихідному порядку, але замінені на інші за невідомим правилом.

Шифр простої заміни

Текст програми:

#include <iostream>

#include <string>

#include <Windows.h>

using namespace std;

string ZLP[33] = { "а","б","в","г","д","е","ё","ж","з","и","й","к","л","м","н","о","п","р","с","т","у","ф","х","ц","ч","ш","щ","ъ","ы","ь","э","ю","я" };

void main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

setlocale(LC\_ALL, "Rus");

cout << "\n КОДИРОВКА \n";

string OPA[33];

int j, n;

cout << "Введите сдвиг: ";

cin >> n;

for (int i = 0; i < 33; i++)

{

j = i + n;

if (j >= 33) { j = i - (33 - n); }

OPA[i] = ZLP[j];

// cout << ZLP[i] << " ";

}

cout << "\nПолученный алфавит:";

cout << "\n";

for (int i = 0; i < 33; i++)

{

cout << OPA[i] << " ";

}

string zhopa, finalOCHKA;

cout << "\n\nТеперь введите сообщение, которое хотите закодировать(без пробелов, нижгим регистром): ";

cin >> zhopa;

for (int i = 0; i < size(zhopa); i++)

{

for (int i1 = 0; i1 < 33; i1++)

{

if (zhopa.substr(i, 1) == ZLP[i1]) { finalOCHKA.append(OPA[i1]); }

}

}

//cout << zhopa.substr(1, 1);

cout << " Закодированое сообщение: " << finalOCHKA << "\n\n";

// раскодировочка

system("pause"); system("cls");

cout << "\n РАСКОДИРОВКА \n";

string Azhopa, fOCHKA;

cout << "\nТеперь введите слово, которое хотите раскодировать: ";

cin >> Azhopa;

for (int i = 0; i < size(Azhopa); i++)

{

for (int i1 = 0; i1 < 33; i1++)

{

if (Azhopa.substr(i, 1) == OPA[i1]) { fOCHKA.append(ZLP[i1]); }

}

}

cout << " Раскодированое сообщение: " << fOCHKA << "\n\n";

}

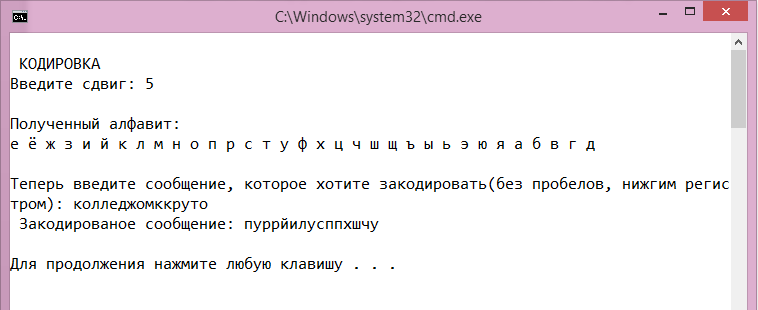
Повідомлення для шифрування: «колледжомккруто»

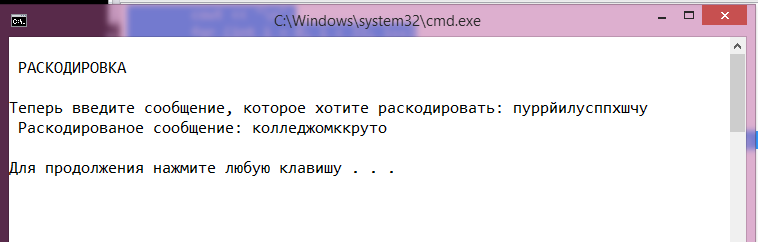
Ключ: зсув алфавіту на 5 позицій вліво.

Зашифроване повідомлення: пуррйлусппхшчу.

Розшифроване повідомлення: колледжомккруто.

Скриншот виконання:





Квадрат Полібія

Текст програми:

#include <iostream>

#include <string>

#include <Windows.h>

using namespace std;

string ZLP[32] = { "а","б","в","г","д","е","ж","з","и","й","к","л","м","н","о","п","р","с","т","у","ф","х","ц","ч","ш","щ","ъ","ы","ь","э","ю","я" };

bool GZH(string ZLP, string pns)

{

bool A = false;

for (int i = 0; i < size(pns); i++)

{

if (ZLP == pns.substr(i, 1))

{

A = true;

}

}

return A;

}

void main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

setlocale(LC\_ALL, "Rus");

string ZHOPA[8][4], pns;

int z = 0;

cout << "Введите ключ (ДО 8 СИМВОЛОВ НИЖНИМ РЕГИСТРОМ): ";

cin >> pns;

//нули

for (int i = 0; i < 8; i++)

{

for (int i1 = 0; i1 < 4; i1++)

{

ZHOPA[i][i1] = "0";

}

}

//ключ

for (int i = 0; i < 8; i++)

{

for (int i1 = 0; i1 < 4; i1++)

{

if (i1 < size(pns)) { ZHOPA[0][i1] = pns.substr(i1, 1); }

if ((i > 0) & ((i1 + 4) < size(pns))) { ZHOPA[1][i1] = pns.substr(i1 + 4, 1); }

}

}

//ПОЛУЧЕННАЯ ТАБЛИЦА

cout << "\nПолученная таблица: \n\n";

for (int i = 0; i < 8; i++)

{

for (int i1 = 0; i1 < 4; i1++)

{

if (ZHOPA[i][i1] == "0")

{

while (GZH(ZLP[z], pns) == true)

{

z++;

}

ZHOPA[i][i1] = ZLP[z];

z++;

}

cout << ZHOPA[i][i1] << " ";

}

cout << endl;

}

string blya, kod, pom;

cout << "\nШо вы хотите закодировать?(НИЖНИМ РЕГИСТРОМ): "; cin >> blya;

for (int j = 0; j < size(blya); j++)

{

for (int i = 0; i < 8; i++)

{

for (int i1 = 0; i1 < 4; i1++)

{

if (ZHOPA[i][i1] == blya.substr(j, 1))

{

if ((i - 1) < 0) { z = 7; pom = ZHOPA[z][i1]; }

else { pom = ZHOPA[i - 1][i1]; }

kod.append(pom);

}

}

}

}

cout << "\nКод: " << kod << endl;

// раскодировочка

system("pause"); system("cls");

cout << "\nА МОЖЕТ ХОТИТЕ ЧТОТО РАСКОДИРОВАТЬ?\n";

string OCHKO, Rochka;

cout << "\nВведите зашифрованное слово: "; cin >> OCHKO;

for (int j = 0; j < size(OCHKO); j++)

{

for (int i = 0; i < 8; i++)

{

for (int i1 = 0; i1 < 4; i1++)

{

if (ZHOPA[i][i1] == OCHKO.substr(j, 1))

{

if ((i + 1) >= 8) { z = 0; pom = ZHOPA[z][i1]; }

else { pom = ZHOPA[i + 1][i1]; }

Rochka.append(pom);

}

}

}

}

cout << "\nРасшифрованное послание: " << Rochka << endl;

cout << endl;

}

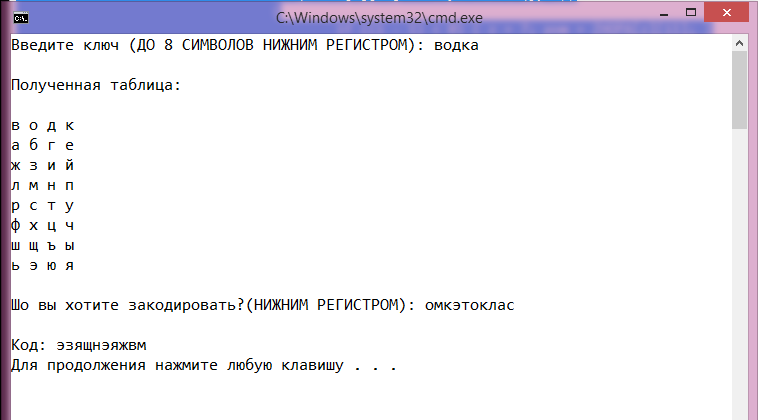
Повідомлення для шифрування: «омкэтоклас».

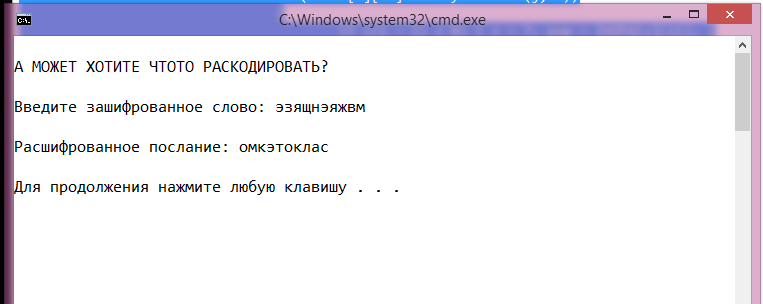
Ключ: водка.

Зашифроване повідомлення: эзящнэяжвм.

Розшифроване повідомлення: омкэтоклас.

Скриншот виконання:





Шифр Тритемія

Текст програми:

#include <iostream>

#include <string>

#include <Windows.h>

using namespace std;

string encrypt(string text, string key)

{

string result = text;

int i = 0;

for (auto& c : result)

{

if (islower(c))

{

c = (((c - 'a') + (key[i++ % key.size()] - 'a')) % 26) + 'a';

}

else if (isupper(c))

{

c = (((c - 'A') + (key[i++ % key.size()] - 'A')) % 26) + 'A';

}

}

return result;

}

string decrypt(string str, string key)

{

string k = key;

for (auto& c : k)

{

c = tolower(c) - 'a' - 1;

c = 'z' - c;

}

return encrypt(str, k);

}

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

setlocale(LC\_ALL, "Rus");

string text, key;

cout << " Ведите сообщение(англ): ";

getline(cin, text);

cout << "\n Ключ: ";

getline(cin, key);

string cipher(encrypt(text, key));

cout << " \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n Зашифровано: " <<cipher << endl;

cout << "\n Расшифровано: " << decrypt(cipher, key) << endl;

}

Повідомлення для шифрування: «kolledzomk».

Ключ: pivo.

Зашифроване повідомлення: zwgztlucbs.

Розшифроване повідомлення: kolledzomk.

Скриншот виконання:

