Министерство науки и образования Украины

Харьковский национальный университет радиоэлектроники

Кафедра Бикс

Отчет

По лабораторной работе № 6:

Тема: «Работа с потоками ввода-вывода. Манипуляторы»

Выполнила: Принял:

Ст. гр БИКС-14-1 Олешко О.И

Д.Т.Ф. Тхао

Харьков 2016

#include <iostream>

#include <string>

#include <vector>

#include <time.h>

using namespace std;

string AlgXOR(string M, string K)

{

int n = 0; string C;

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

C.push\_back(M[i] ^ K[i]);

n += (n == K.size()) ? -n : 1;

}

return C;

}

string AlgSubst(string M, string K)

{

string C;

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

C.push\_back(K[M[i] % K.size()]);

}

return C;

}

void GenKeyPolySubst(string \*Ks, string K)

{

int a;

srand(time(0));

for (int i = 0; i<3; i++){

vector<int> b; b.resize(K.size());

for (auto j : b) {

do {

a = rand() % K.size(); b[a]++;

} while (b[a] > 1);

Ks[i].push\_back(K[a]);

} cout << "Key[" << i << "] = " << Ks[i] << endl;

b.clear();

}

}

string AlgPolySubst(string M, string K)

{

string C, Keys[3];

GenKeyPolySubst(Keys, K);

for (int i = 0; i<M.size(); i++) {

C.push\_back(Keys[i % 3][M[i] % K.size()]);

}

return C;

}

class Crypt {

int alg\_c;

string Key\_c;

ostream \*os\_c;

public:

Crypt(int alg, string Key);

void set\_os(ostream \*os);

string Encryption(const char\*M);

friend Crypt& operator<<(ostream &os, Crypt &C);

friend Crypt& operator<< (Crypt &c, const char \*M);

friend Crypt& operator<< (Crypt &c, int A);

};

string Crypt::Encryption(const char\*M)

{

switch (this->alg\_c) {

case 1: return AlgXOR(M, this->Key\_c);

case 2: return AlgSubst(M, this->Key\_c);

case 3: return AlgPolySubst(M, this->Key\_c);

}

}

Crypt::Crypt(int alg, string Key) {

alg\_c = alg;

Key\_c = Key;

os\_c = 0;

}

Crypt encrypt(int a, string K) {

Crypt C(a, K);

return C;

};

void Crypt::set\_os(ostream \*os) {

this->os\_c = os;

}

Crypt& operator<< (ostream &os, Crypt &C)

{

C.set\_os(&os);

return C;

}

Crypt& operator<< (Crypt &c, const char \*M) {

\*c.os\_c << "\nEncrypt: " << c.Encryption(M) << "\n\n";

return c;

}

Crypt& operator<< (Crypt &c, int A) {

string M;

while (A) {

M.push\_back(A | 0xf);

A >>= 8;

}

\*c.os\_c << "Encrypt: " << c.Encryption(M.c\_str()) << "\n\n";

return c;

}

int main()

{

int a;

string Key;

cout << "Algorithm: ";

cin >> a; cin.get();

cout << "Key: ";

getline(cin, Key);

cout << encrypt(a, Key) << "Message" << 12345;

system("pause");

return 0;

}





