МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ

УНИВЕРСИТЕТ им. Р.Е.АЛЕКСЕЕВА

Институт радиоэлектроники и информационных технологий

ОТЧЕТ

по лабораторной работе №2

«Шифрование методом гаммирования»

по дисциплине

«Методы и средства защиты информации»

РУКОВОДИТЕЛЬ:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_ Капранов С. Н.\_\_

(подпись) (фамилия, и.,о.)

СТУДЕНТ:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Сухоруков В.А.\_\_\_

(подпись) (фамилия, и.,о.)

\_\_\_\_\_\_19-ВМ\_\_\_\_\_\_\_\_

(шифр группы)

Работа защищена «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_

С оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Нижний Новгород 2022

# Задание №2.2

# Реализовать алгоритм шифрования данных «Шифрование методом гаммирования» по модулю N.

# Описание алгоритма.

# Символы исходного сообщения заменяются числами, которые складываются по модулю с числами гаммы. Ключом шифра является гамма, символы которой последовательно повторяются.

# Перед шифрованием символы сообщения и гаммы заменяются их номерами в алфавите и само кодирование выполняется по формуле

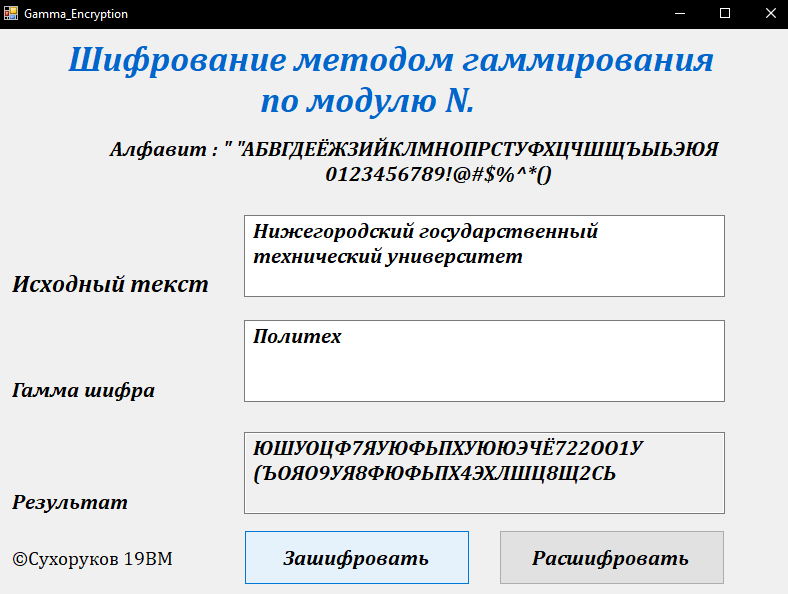
***Ci = (Mi+Ki) mod N***

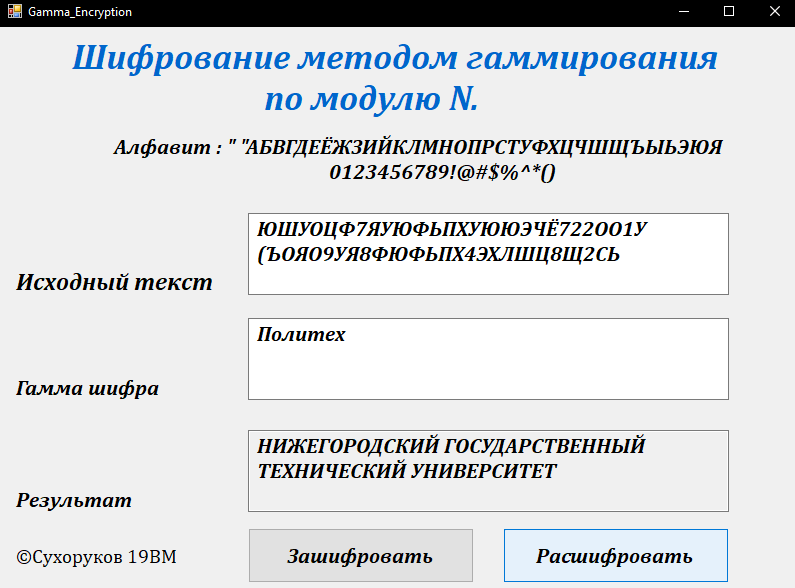
Расшифровывание:

***Mi = (Ci-Ki+N) mod N***

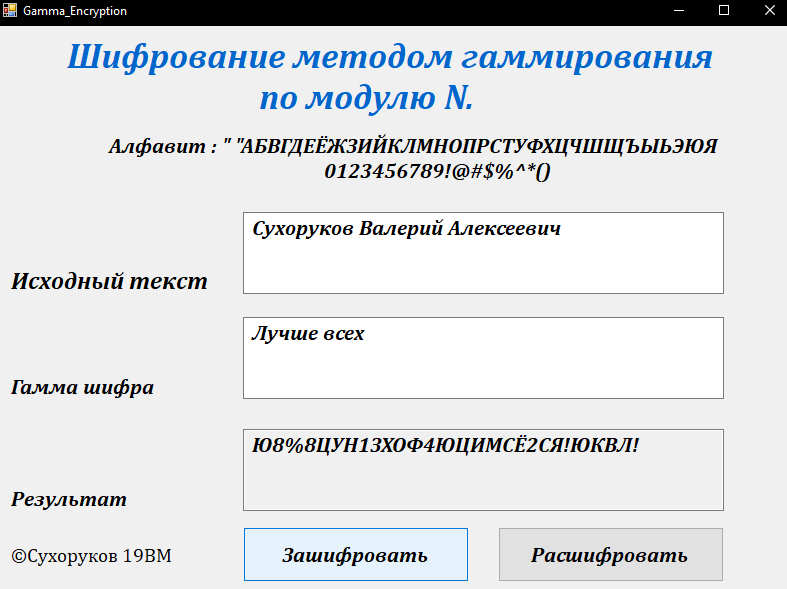
# Примеры шифрования и расшифровывания

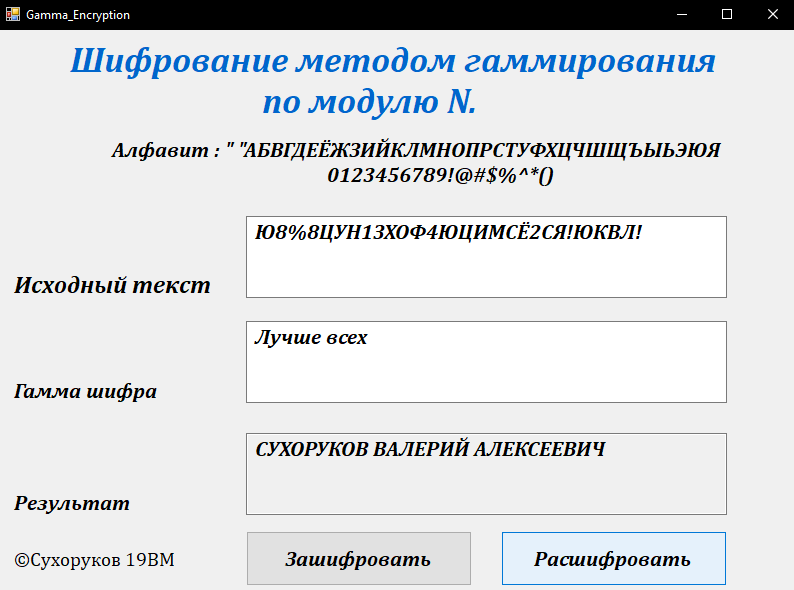
* Нижегородский государственный технический университет





* Сухоруков Валерий Алексеевич





# Текст программы

* Функция зашифровки текста

//Зашифровать

String^ Encode(String^ input\_string, String^ key, string language) {

String^ characters(L"");

characters =

" АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ0123456789!@#$%^&\*()";

size\_t N = characters->Length;

String^ res="";

size\_t key\_index = 0,c=0,c1=0,c2=0;

//Приведение букв открытого текста и ключа

//к верхнему регистру

key = key->ToUpper();

input\_string = input\_string->ToUpper();

for (size\_t i = 0; i < input\_string->Length; i++) {

//Поиск в алфавите текущей буквы открытого текста

//И запоминание её номера

c = characters->IndexOf(input\_string[i]);

// Поиск в алфавите текущей буквы ключа

//И запоминание её номера

c1= characters->IndexOf(key[key\_index]);

//Расчет номера буквы закрытого текста

c2 = (c1 + c) % N;

//Добавление буквы в зашифрованный текст

res = res + characters[c2];

key\_index++;

//Если индекс буквы ключа равен длине ключа,

//То повторяем ключ

if (key\_index == key->Length) { key\_index = 0; }

}

return res;

}

* Функция расшифровки текста

//Расшифровать

String^ Decode(String^ input\_string, String^ key,string language) {

String^ characters(L"");

characters =

" АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ0123456789!@#$%^&\*()";

size\_t N = characters->Length;

String^ res = "";

size\_t key\_index = 0, c = 0, c1 = 0, c2 = 0;

//Приведение букв закрытого текста и ключа

//к верхнему регистру

key = key->ToUpper();

input\_string = input\_string->ToUpper();

for (size\_t i = 0; i < input\_string->Length; i++) {

//Поиск в алфавите текущей буквы закрытого текста

//И запоминание её номера

c = characters->IndexOf(input\_string[i]);

//Поиск в алфавите текущей буквы ключа

//И запоминание её номера

c1 = characters->IndexOf(key[key\_index]);

//Расчет номера буквы открытого текста

c2 = (c - c1 + N) % N;

//Добавление буквы в расшифрованный текст

res = res + characters[c2];

key\_index++;

//Если индекс буквы ключа равен длине ключа,

//То повторяем ключ

if (key\_index == key->Length) { key\_index = 0; }

}

return res;

}