ФГБОУ ВО «УФИМСКИЙ УНИВЕРСИТЕТ НАУКИ И ТЕХНОЛОГИЙ»

Факультет Информатики и робототехники

Кафедра Вычислительной математики и кибернетики

Направление 09.03.04 «Программная инженерия»

**КУРСОВОЙ ПРОЕКТ**

**по дисциплине «Основы микропрограммирования»**

НА ТЕМУ:«Реализация программы шифрования и дешифрования данных на языке Ассемблер.»

Выполнили студенты гр. ПРО-231Б:

Семенов Л.А. (\_\_\_\_\_\_\_\_\_\_\_\_\_)

(фамилия, инициалы) (подпись)

Семенов Г.А. (\_\_\_\_\_\_\_\_\_\_\_\_\_)

(фамилия, инициалы) (подпись)

Проверила доцент каф. ВМиК

Нургаянова О.С. (\_\_\_\_\_\_\_\_\_\_\_\_\_)

(фамилия, инициалы) (подпись)

Уфа 2022 г.

**Аннотация**

Цель и задачи курсовой работы определили её структуру. Она состоит из введения, двух разделов, заключения и списка литературы.

В первом разделе рассмотрена история появления и развития методов шифрования.

Второй раздел посвящён разработке программного продукта, в нём описывается структура программы и проведены тестирования программы и анализ полученных результатов.

Заключение содержит в себе результаты проделанной работы при реализации методов шифрования.

Представлены блок-схемы, описывающие структуры программы, и листинг кода разработанного программного продукта.

# Содержание

Введение 4

Теория 5

# 1. Исследование и анализ предметной области 5

## 1.1 История появления и развития методов шифрования 5

## 1.2 Методы шифрования 6

Реализация программы 10

Блок-схемы программы 11

Код программы 13

Скриншоты работы программы 20

Заключение 22

Список использованной литературы 23

# Введение

Шифрование — это преобразование информации, делающее ее нечитаемой для посторонних. При этом доверенные лица могут провести дешифрование и прочитать исходную информацию. Существует множество способов шифрования / дешифрования, но секретность данных основана не на тайном алгоритме, а на том, что ключ шифрования (пароль) известен только доверенным лицам.

Шифрование появилось около четырех тысяч лет тому назад. Первым известным примером шифра считается египетский текст, созданный примерно в 1900 г. до н.э., в котором вместо обычных для египтян иероглифов использовались не совпадающие с ними символы.

На компьютере можно зашифровать абсолютно любой объект. Так как изучение криптографии началось еще в древности, то изучение шифров зашло уже очень далеко. Но проблема в том, что как только шифр создается, очень быстро находят способ дешифровки.

**Актуальность курсовой работы** - благодаря курсовой работе я освоила новые, для меня, методы, функции и конструкции в Turbo Pascal, а также расширила свой кругозор в области шифрования текста.

**Объект курсовой работы** — это создание программного продукта.

**Предмет курсовой работы** — это программный продукт, реализующий такие методы шифрования как шифр Цезаря, шифр с заменой букв цифрами и шифр Полибиуса.

**Цель курсовой работы** - создание программного продукта, который бы позволял обычным пользователям компьютера зашифровывать и дешифровать необходимый им текст.

Для достижения этой цели были поставлены следующие задачи:

 изучить теоретические сведения, необходимые для решения данной задачи;

 систематизировать и обобщить полученные знания;

 создать пользовательский интерфейс;

 создать программный продукт реализующий, реализующий три шифра.

 Провести тестирование и отладку программы.

# 1. Исследование и анализ предметной области

## 1.1 История появления и развития методов шифрования

Люди издавна использовали шифрование - как способ защиты информации. Идея скрыть в тексте тайные значения и сообщения почти так же стара, как и само искусство письма.

На заре бронзового века писцов Ближнего Востока обучали искусству расшифровки запутанных и закодированных текстов.

Древние писцы разгадывали гораздо более сложные загадки. Наиболее популярной из них было использование акростиха, когда скрытое сообщение прочитывали, сложив вместе первые буквы каждой строки текста. За века своего существования человечество придумало множество способов хранения тайны.

Степень изученности методов шифрования достаточно высокая. Каждый год создается множество программ, литературы, посвящённым криптографическим системам. Криптографическая система - семейство преобразований шифра и совокупность ключей. Существуют Симметричные и Асимметричные криптосистемы.

Симметричные криптосистемы (с секретным ключом **- secret key** systems**) -** данные криптосистемы построены на основе сохранения в тайне ключа шифрования. Процессы зашифрования и расшифрования используют один и тот же ключ. Секретность ключа является постулатом.

Асимметричные криптосистемы (системы открытого шифрования - соткрытым ключом public key systems) - смысл данных криптосистем состоит в том, что для зашифрования и расшифрования используются разные преобразования. Одно из них - зашифрование - является абсолютно открытым для всех. Другое же - расшифрование - остается секретным.

На данный момент чаще используются достаточно стойкие системы, системы с достаточно сложным алгоритмом шифрования. Из-за необходимости различных объектов зашифровывать секретные данные и криптографические системы не стоят на месте и постоянно совершенствуются.



Рис. 1.1 Модель процесса шифрования и дешифрования

## 1.2 Методы шифрования

Самым простым шифром является **ШИФР С ЗАМЕНОЙ БУКВ ЦИФРАМИ**. Каждой букве соответствует число по алфавитному порядку. А-1, B-2, C-3 и т.д.

**Книжный шифр**

В таком шифре ключом является некая книга, имеющаяся и у отправителя, и у получателя. В шифре обозначается страница книги и строка, первое слово которой и является разгадкой. Дешифровка невозможна, если книги у отправителя и корреспондента разных годов издания и выпуска. Книги обязательно должны быть идентичными.

**Шифр с кодовым словом**

Еще один простой способ как в шифровании, так и в расшифровке. Используется кодовое слово (любое слово без повторяющихся букв). Данное слово вставляется впереди алфавита и остальные буквы по порядку дописываются, исключая те, которые уже есть в кодовом слове. Пример: кодовое слово - NOTEPAD.

Исходный: A B C D E F G H I J K L M N O P Q R S T U V W X Y Z  
Замена: N O T E P A D B C F G H I J K L M Q R S U V W X Y Z

**Цифровой таблицы**

Её параметры могут быть какими угодно, главное, чтобы получатель и отправитель были в курсе. Пример цифровой таблицы.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 |
| 1 | A | B | C | D | E |
| 2 | F | G | H | I | J |
| 3 | K | L | M | N | O |
| 4 | P | Q | R | S | T |
| 5 | U | V | W | X | Y |
| 6 | Z | 1 | 2 | 3 | 4 |
| 7 | 5 | 6 | 7 | 8 | 9 |
| 8 | 0 | . | , | ? | ! |

Таблица 1.1. Цифровая таблица

Сложность дешифрования заключается в определении шифра. Как только он определен, сообщение легко раскладывается по алфавиту.



Рис. 1.2 Масонский шифр

**Шифр цезаря**

Он является историческим примером шифра замены (1 век до н.э.), описанным историком Древнего Рима Светонием. Гай Юлий Цезарь использовал в своей переписке шифр собственного изобретения. Применительно к современному русскому языку он состоял в следующем. Выписывался алфавит, затем под ним выписывался тот же алфавит, но с циклическим сдвигом на 3 буквы влево:

АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЫЪЬЭЮЯ

ГДЕЁЖЗИЙКЛМНОП РСТУФХЦ ЧШ ЩЫЪЬ ЭЮЯАБВ

**Математическая модель**

Если сопоставить каждому символу алфавита его порядковый номер (нумеруя с 0), то шифрование и дешифрование можно выразить формулами:





где x - символ открытого текста, - символ шифрованного текста, - мощность алфавита (кол-во символов),- ключ.

Теоретический анализ литературы позволяет выделить перспективное направление разработки методов шифрования. В работе предлагается создание программы с использованием трёх методов шифрования: шифр Цезаря, шифр с заменой букв цифрами и шифр Полибиуса. Все эти шифры достаточно просты для понимания и дешифрования, но в этом то и заключается цель - создание пользовательской программы для развития детей.

Пользователь будет вводить слово или предложение, которое ему хотелось бы зашифровать. Затем, используя метод шифрования будет вручную расшифровывать, а после того как получит результат - проверит его c помощью программы.

### Шифр Полибия

Система Цезаря не является старейшей. Возможно, что наиболее древней из известных является система греческого историка Полибия, умершего за 30 лет до рождения Цезаря. Его суть состоит в следующем: рассмотрим прямоугольник, часто называемый доской Полибия.

### Методы вскрытия одноалфавитных систем

При своей простоте в реализации одноалфавитные системы легко уязвимы. Определим количество различных систем в аффинной системе. Каждый ключ полностью определен парой целых чисел a и b, задающих отображение ax+b. Для, а существует j(n) возможных значений, где j(n) - функция Эйлера, возвращающая количество взаимно простых чисел с n, и n значений для b, которые могут быть использованы независимо от a, за исключением тождественного отображения (a=1 b=0), которое мы рассматривать не будем. Таким образом получается j(n)\*n-1 возможных значений, что не так уж и много: при n=33 в качестве a могут быть 20 значений (1, 2, 4, 5, 7, 8, 10, 13, 14, 16, 17, 19, 20, 23, 25, 26, 28, 29, 31, 32), тогда общее число ключей равно 20\*33-1=659. Перебор такого количества ключей не составит труда при использовании компьютера. Но существуют методы, упрощающие этот поиск и которые, могут быть использованы при анализе более сложных шифров.

## Многоалфавитные системы

Полиалфавитные подстановочные шифры были изобретены Лином Баттистой (Lean Battista) в 1568 году. Основная идея многоалфавитных систем состоит в том, что на протяжении всего текста одна и та же буква может быть зашифрована по-разному. Т.е. замены для буквы выбираются из многих алфавитов в зависимости от положения в тексте. Это является хорошей защитой от простого подсчета частот, так как не существует единой маскировки для каждой буквы в криптотексте. В данных шифрах используются множественные однобуквенные ключи, каждый из которых используется для шифрования одного символа открытого текста. Первым ключом шифруется первый символ открытого текста, вторым - второй, и т.д. После использования всех ключей они повторяются циклически.

### Шифр Вернама

Шифр Вернама, или одноразовый блокнот, был изобретен в 1917 году Мейджором Джозефом Моборном (Major Joseph Mauborn) и Гильбертом Вернамом (Gilbert Vernam) из AT&T (American Telephone & Telegraph). В классическом понимании одноразовый блокнот является большой неповторяющейся последовательностью символов ключа, распределенных случайным образом. Первоначально это была одноразовая лента для телетайпов. Отправитель использовал каждый символ ключа для шифрования только одного символа открытого текста. Шифрование представляет собой сложение по модулю n (мощность алфавита) символа открытого текста и символа ключа из одноразового блокнота. Каждый символ ключа используется только один раз и для единственного сообщения, иначе даже если использовать блокнот размером в несколько гигабайт, при получении криптоаналитиком нескольких текстов с перекрывающимися ключами он сможет восстановить исходный текст. Он сдвинет каждую пару шифротекстов относительно друг друга и подсчитает число совпадений в каждой позиции. если шифротексты смещены правильно, соотношение совпадений резко возрастет. С этой точки зрения криптоанализ не составит труда. Если же ключ не повторяется и случаен, то криптоаналитик, перехватывает он тексты или нет, всегда имеет одинаковые знания. Случайная ключевая последовательность, сложенная с неслучайным открытым текстом, дает совершенно случайный криптотекст, и никакие вычислительные мощности не смогут это изменить.

В реальных системах сначала подготавливают две одинаковые ленты со случайными цифрами ключа. Одна остается у отправителя, а другая передается «неперехватываемым» образом, например, курьером с охраной, законному получателю. Когда отправитель хочет передать сообщение, он сначала преобразует его в двоичную форму и помещает в устройство, которое к каждой цифре сообщения прибавляет по модулю два цифры, считанные с ключевой ленты. На принимающей стороне кодированное сообщение записывается и пропускается через машину, похожую на устройство, использованное для шифрования, которое к каждой двоичной цифре сообщения прибавляет (вычитает, так как сложение и вычитание по модулю два эквивалентны) по модулю два цифры, считанные с ключевой ленты, получая таким образом открытый текст. При этом, естественно, ключевая лента должна продвигаться абсолютно синхронно со своим дубликатом, используемым для зашифрования.

Главным недостатком данной системы является то, что для каждого бита переданной информации должен быть заранее подготовлен бит ключевой информации, причем эти биты должны быть случайными. При шифровании большого объема данных это является серьезным ограничением. Поэтому данная система используется только для передачи сообщений наивысшей секретности. По слухам «горячая линия» между США и СССР шифровалась с помощью одноразового блокнота. Многие сообщения советских шпионов были зашифрованы с использованием одноразовых блокнотов. Эти сообщения нераскрыты сегодня, и не будут раскрыты никогда (если не найдется способа вернуться в прошлое и достать эти блокноты

## Обзор существующих программ

На данный момент программы для зашифровки текстом пока еще не использовались, с развивающей целью, но программ шифрующих информацию с целью секретности существует о множество:Word - бесплатная программа для шифрования и расшифровки текстов. Это надежное и простое в использовании приложение позволяет быстро зашифровать и расшифровать текст с помощью пароля. - программа для защиты файлов. С помощью этой программы можно защитить папки и файлы функцией двойной идентификации пользователя. - это бесплатная программа которая позволяет шифровать данные в операционных системах Windows Vista/XP/Seven, Mac OS X и Linux.Steganography - программа. которая поможет скрыть текст или файлы внутри изображений.

Программ для шифрования файлов существует бессчетное множество, большинство из них просты и удобны в применении. Каждому пользователю, беспокоящемуся о сохранности своих данных, необходимо знать о таких программах и пользоваться ими.

**Реализация программы**

## Требования к программному продукту

При разработке программного продукта требовалось решить следующие задачи:

. Формирование меню,

. Формирование процедур, реализующих методы шифрования,

. Вывод результата на экран,

. Остановка программы и выход.

Данная программа реализует два метода шифрования, поэтому для простоты выбора метода необходимо было создать удобный интерфейс для пользователя.

Программа должна иметь простую систему управления. Поэтому в меню использованы максимально лаконичные команды и ответы на них. Это во многом облегчает работу пользователя.

Суть работы подпрограммы заключается в том, что на экран последовательно выводятся команды:

Для начала пользователь должен выбрать метод, с помощью которого он хочет зашифровать текст. На экран выводится - «Выберете способ шифрования 1-шифр Цезаря, 2-шифр с заменой букв цифрами»

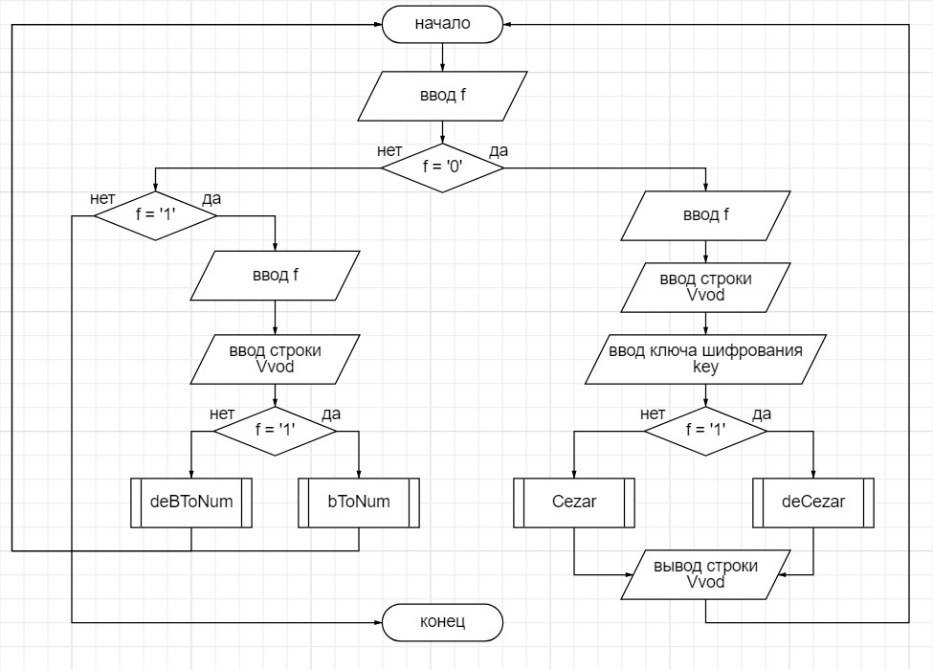
Пользователь выбирает, нажимая номер нужного ему шифра.

Если же он не хочет шифровать или уже закончил занятие, тогда для выхода из цикла ему следует нажать 0. Сообщение с этой информацией выводится на экран вместе со всеми предыдущими «Для выхода нажмите 0».

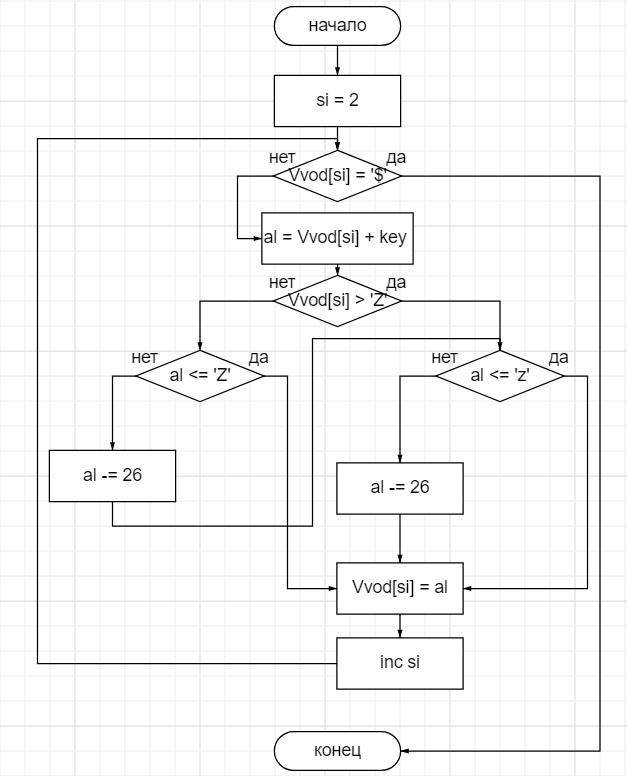
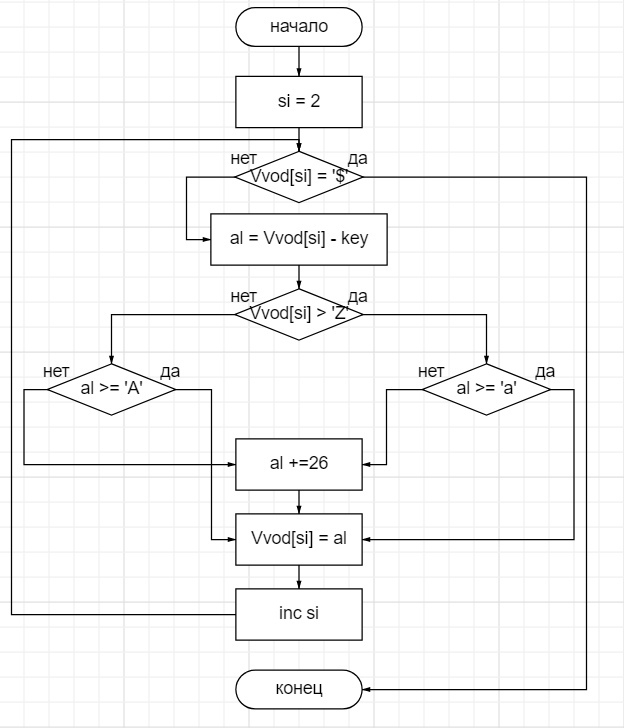
Меню управляется клавиатурой: для выбора следует нажать требуемый номер и ENTER.

После запуска программного продукта на экране появляется меню, состоящее из предложения выбора шифра и выхода из программы.

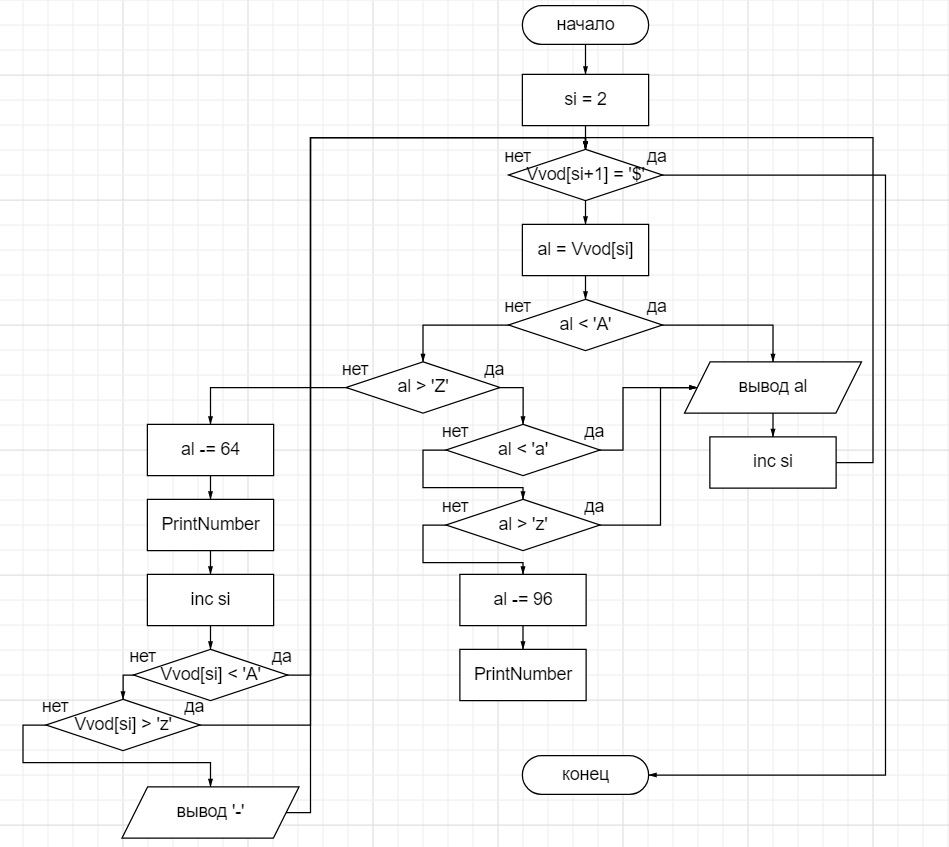
**Блок-схемы программы**



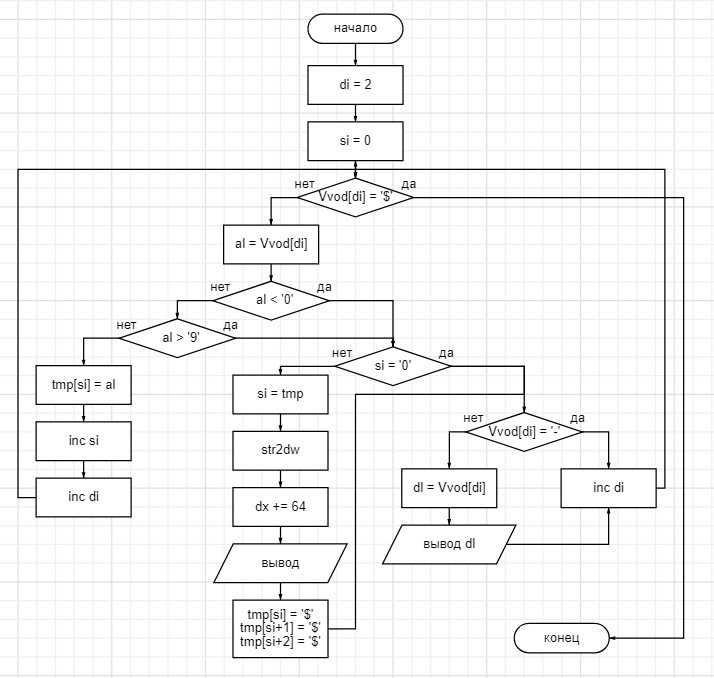
Блок-схема: main программы

Блок-схема: cezar Блок-схема: deCezar



Блок-схема: bToNum



Блок-схема: debToNum

**Код программы**

.model small  
.stack 256H  
.data  
mes1 db 10, 13, 10, 13, '0. Cezar',10,13 ,'1. bToNum',10,13, '2. Exit $'  
mes db 10, 13, 10, 13,'Enter message: $'  
mes2 db 10, 13,'Enter key: $'  
mes3 db 10, 13,10, 13,'0. Encrypt, 1. Decrypt: $'  
msg1 db 10, 13, 'Result: $'  
f db 6 dup('$')  
mes30 db 10, 13,10, 13,'0. Encrypt, 1. Decrypt: $'  
vvod db 50 dup('$')  
mes31 db 10, 13,10, 13,'0. Encrypt, 1. Decrypt: $'  
ent db 13, 10, '$'  
key db 0  
keystr db 6 dup('$')  
tmp db 6 dup('$')  
  
.code  
  
bToNum proc; Это функция шифрования из букв в числа, A=1, B=2 и т.д.  
mov si, 2  
  
  
m1:  
xor ah, ah  
cmp vvod[si+1], '$'; если встречаем конец строки  
je m0; то возращаемся из процедуры  
  
  
mov al, vvod[si]; кидаем в ал символ  
cmp al, 'A'  
jl m4; если символ меньше чем А, то прыгаем на М4  
cmp al, 'Z'  
jg m2; если больше чем Z, то на М2  
  
sub al, 64; вычитаем 64, что превратить в число  
call PrintNumber; выводим  
jmp m3; прыгаем на М3  
  
m2:  
cmp al, 'a'; если меньше чем а  
jl m4; то прыгаем на М4  
cmp al, 'z'; если больше чем z  
jg m4; то на м4  
  
sub al, 96; вычитаем 96  
call PrintNumber; и выводим  
  
m3:  
inc si; увеличиваем регистр, которым проходимся по строке  
cmp vvod[si], 'A';  
jl m30; если меньше чем А, прыгаем на М30  
cmp vvod[si], 'z'; больше чем z  
jg m30; на м30  
mov dl, '-';  
mov ah,02h  
int 21h; выводим на экран '-'  
m30:  
jmp m1  
  
m4:  
mov dl, al  
mov ah,02h  
int 21h;тут выводим символ  
inc si  
jmp m1  
  
m0:  
ret  
bToNum endp  
  
deBToNum proc; функция дешифрования, делает все наоборот, 1=A, 2=B  
mov di, 2  
  
d0:  
mov si, 0  
d1:  
cmp vvod[di], '$'; сравниваем с концом строки  
je d4; если конец - выоходим  
mov al, vvod[di]  
cmp al, '0'; если символ меньше чем "0"  
jl d2; прыгаем на d2  
cmp al, '9'  
jg d2; больше чем 9, то прыгаем на д2  
mov tmp[si], al; добавляем символ во временную строку  
inc di; увеличиваем регистры  
inc si  
jmp d1  
  
d2:  
cmp si, 0; сравниваем si с нулем  
je skip30; если нуль, то мы не ввели не одну цифру, значит выводит ччисло не нужно  
lea si, tmp; кидаем адрес временной строки в си  
call str2dw; превращаем в число  
mov dx, ax  
add dx, 64; добавляем 64, чтобы превратить в букву  
mov ah, 02h  
int 21h; и выводим  
mov si, 0  
mov tmp[si], '$'; очищаем временную строку  
mov tmp[si+1], '$'  
mov tmp[si+2], '$'  
  
skip30:  
cmp vvod[di], '-'  
je skip3  
mov dl, vvod[di]  
mov ah, 02h  
int 21h;выводим символ, если это не "-"  
skip3:  
inc di  
  
jmp d0  
  
d4:  
ret  
deBToNum endp  
  
str2dw proc; функция перевода строки в число  
xor dx,dx ;сумма  
nch: xor ax,ax  
lodsb ;берем cимвол по адресу si в al  
test al,al ;если это нулев байт, то заканчиваем  
jz ex  
cmp al,'9' ;Если это не цифра, то пропускаем  
jnbe nch  
cmp al,'0' ;Если это не цифра, то пропускаем  
jb nch  
sub ax,'0' ;получаем цифровое значение  
shl dx,1 ;умножаем сумму на 10  
add ax, dx  
shl dx, 2  
add dx, ax ;прибавляем текущее значение  
jmp nch  
ex: mov ax,dx  
ret  
str2dw endp  
  
Cezar proc; тут шифруем методом цезаря. Просто добавляем к каждому символу наш ключ, и если вышел символ за границы алфавита, выводим с начала списка алфавита  
push cx  
push bx  
push di  
mov si, 2  
lp1:  
cmp vvod[si], '$'  
je exs  
mov al, vvod[si]  
mov bl, vvod[si]  
add al, key  
cmp bl, 'Z'  
jg skip2  
cmp al, 'Z'  
jle skip1  
sub al, 26  
skip2:  
cmp al, 'z'  
jle skip1  
sub al, 26  
skip1:  
mov vvod[si], al  
inc si  
jmp lp1  
exs:  
pop di  
pop bx  
pop cx  
ret  
Cezar endp  
  
deCezar proc; тут дешифруем методом цезаря  
push cx  
push bx  
push di  
mov si, 2  
deLp1:  
cmp vvod[si], '$'  
je deExs  
mov al, vvod[si]  
mov bl, vvod[si]  
sub al, key  
cmp bl, 'Z'  
jg deSkip2  
cmp al, 'A'  
jge deSkip1  
add al, 26  
jmp deSkip1  
deSkip2:  
cmp al, 'a'  
jge deSkip1  
add al, 26  
deSkip1:  
mov vvod[si], al  
inc si  
jmp deLp1  
deExs:  
pop di  
pop bx  
pop cx  
ret  
deCezar endp  
  
PrintNumber proc ;процедура показа числа из AX  
push BX ;сохраняем содержимое регистров в  
push AX  
push DX  
push SI  
push CX

mov CX, 0 ;инициализируем цикл  
mov BX, 10 ;в BX заносим основание системы  
счисления  
loophere: ;основной цикл  
mov DX, 0 ;обнуляем DX  
div BX ;делим на 10  
push AX ;результат в стеке  
add DL, "0" ;конвертируем последнюю цифру в ASCII-код  
pop AX ;вернем AX  
push DX ;сохраним DX  
inc CX ;увеличим CX на 1  
cmp AX, 0 ;повторяем для всех цифр числа  
jnz loophere  
mov AH, 2 ;DOS-функция вывода символа  
loophere2:  
pop DX ;восстанавливаем цифры от последней  
к первой и  
mov AH, 02h ;выводим их на экран  
int 21h  
loop loophere2  
pop CX  
pop SI  
pop DX  
pop BX  
pop AX  
ret  
PrintNumber endp  
  
start:  
mov ax, [@data](https://vk.com/data)  
mov ds,ax  
  
start0:  
  
mov ah, 9 ;функция 9 прерывания 21h - вывод строки на экран  
lea dx, mes1 ;берём адрес строки  
int 21h ;вызов прерывания для вывода строки  
  
;вводим цифру во флажок  
lea dx, f; кидаем адрес строки в dx  
mov ah, 0ah; прерывание ввода с консоли  
int 21h;тут мы вводим строку с консоли  
  
mov si, 2  
  
cmp f[si], '0'  
je Cezar0; если нуль, то прыгаем на шифрование цезарем  
  
cmp f[si], '1'  
je bToNum0; если 1 - то методом букв в числа  
  
mov ax,4c00h ;завершаем программу  
int 21h  
  
Cezar0:  
  
mov ah, 9 ;функция 9 прерывания 21h - вывод строки на экран  
lea dx, mes3 ;берём адрес строки  
int 21h ;вызов прерывания для вывода строки  
  
;здесь опять вводим цифру во фладок, шифровать или дешифровать  
lea dx, f; кидаем адрес строки в dx  
mov ah, 0ah; прерывание ввода с консоли  
int 21h;тут мы вводим строку с консоли  
  
  
  
mov ah, 9 ;функция 9 прерывания 21h - вывод строки на экран  
lea dx, mes ;берём адрес строки  
int 21h ;вызов прерывания для вывода строки  
  
lea dx, vvod; кидаем адрес строки в dx  
mov ah, 0ah; прерывание ввода с консоли  
int 21h;тут мы вводим строку с консоли  
  
mov si, 1  
xor ah, ah  
mov al, vvod[si]  
add si, ax  
inc si  
mov vvod[si], '$'  
  
mov ah, 9 ;функция 9 прерывания 21h - вывод строки на экран  
lea dx, mes2 ;берём адрес строки  
int 21h ;вызов прерывания для вывода строки  
  
;вводим ключ шифрования  
lea dx, keystr; кидаем адрес строки в dx  
mov ah, 0ah; прерывание ввода с консоли  
int 21h;тут мы вводим строку с консоли  
  
lea si, keystr  
call str2dw;и превращаем наш ключ в число, так как ввели его в строковом формате  
mov key, al  
  
  
  
mov si, 2  
cmp f[si], '1'  
je deCrypt; если ввели 1 - то прыгаем на дешифрование  
  
call Cezar; иначе вызываем шифрование цезарем  
jmp vivod; прыгаем к выводу  
  
deCrypt:  
call deCezar; вызываем дешифрование  
  
vivod:  
mov ah, 9 ;функция 9 прерывания 21h - вывод строки на экран  
lea dx, msg1 ;берём адрес строки  
int 21h ;вызов прерывания для вывода строки  
  
mov ah, 9 ;функция 9 прерывания 21h - вывод строки на экран  
lea dx, vvod ;берём адрес строки  
inc dx  
inc dx  
int 21h ;вызов прерывания для вывода строки  
  
jmp start0; и возвращаемся к выбору вариантов  
  
bToNum0:  
mov ah, 9 ;функция 9 прерывания 21h - вывод строки на экран  
lea dx, mes3 ;берём адрес строки  
int 21h ;вызов прерывания для вывода строки  
  
lea dx, f; кидаем адрес строки в dx  
mov ah, 0ah; прерывание ввода с консоли  
int 21h;тут мы вводим строку с консоли  
  
mov ah, 9 ;функция 9 прерывания 21h - вывод строки на экран  
lea dx, mes ;берём адрес строки  
int 21h ;вызов прерывания для вывода строки  
  
lea dx, vvod; кидаем адрес строки в dx  
mov ah, 0ah; прерывание ввода с консоли  
int 21h;тут мы вводим строку с консоли  
  
  
mov si, 2  
cmp f[si], '1'  
je deCrypt0  
  
mov ah, 9 ;функция 9 прерывания 21h - вывод строки на экран  
lea dx, msg1 ;берём адрес строки  
int 21h ;вызов прерывания для вывода строки  
  
call bToNum  
  
jmp start0  
  
  
deCrypt0:  
  
mov ah, 9 ;функция 9 прерывания 21h - вывод строки на экран  
lea dx, msg1 ;берём адрес строки  
int 21h ;вызов прерывания для вывода строки  
  
call deBToNum  
  
jmp start0  
  
ext:  
mov ax,4c00h ;завершаем программу  
int 21h  
  
end start

**Скриншоты работы программы.**

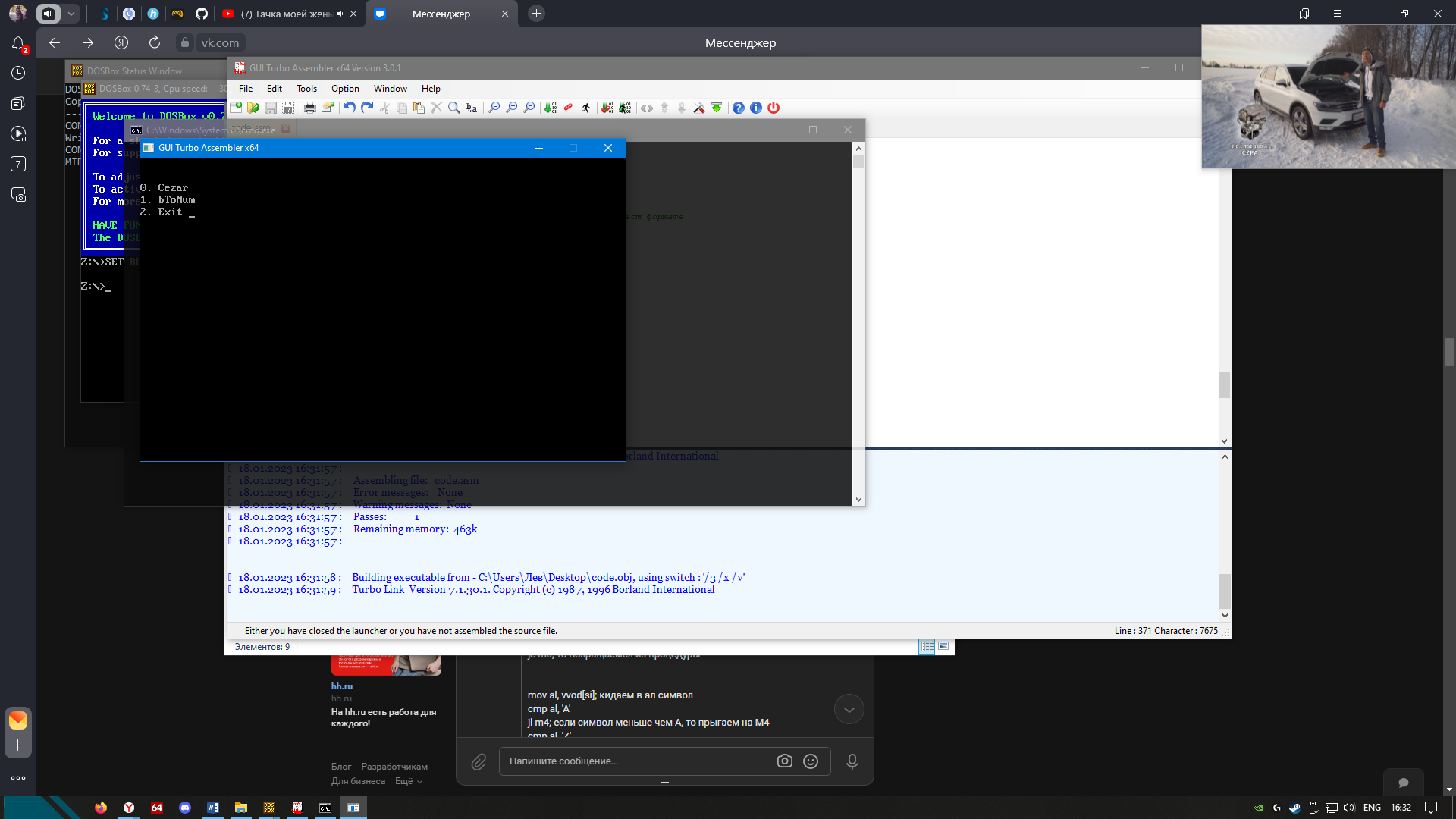


Рис.2 Запуск программы

После выбора шифра выводится еще одно предложение - зашифровать или расшифровать текст.

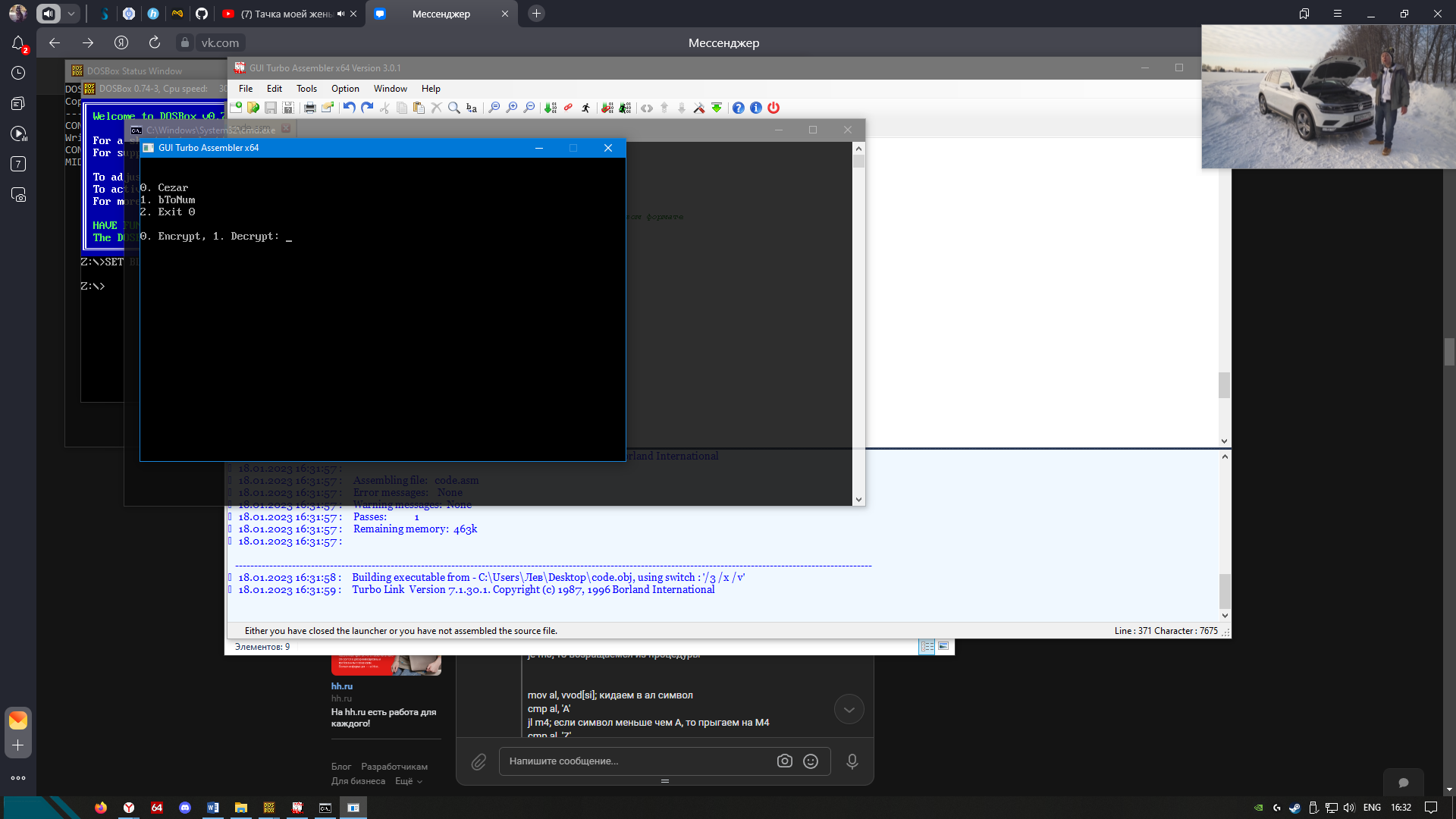


Рис.3 Выбор процедуры

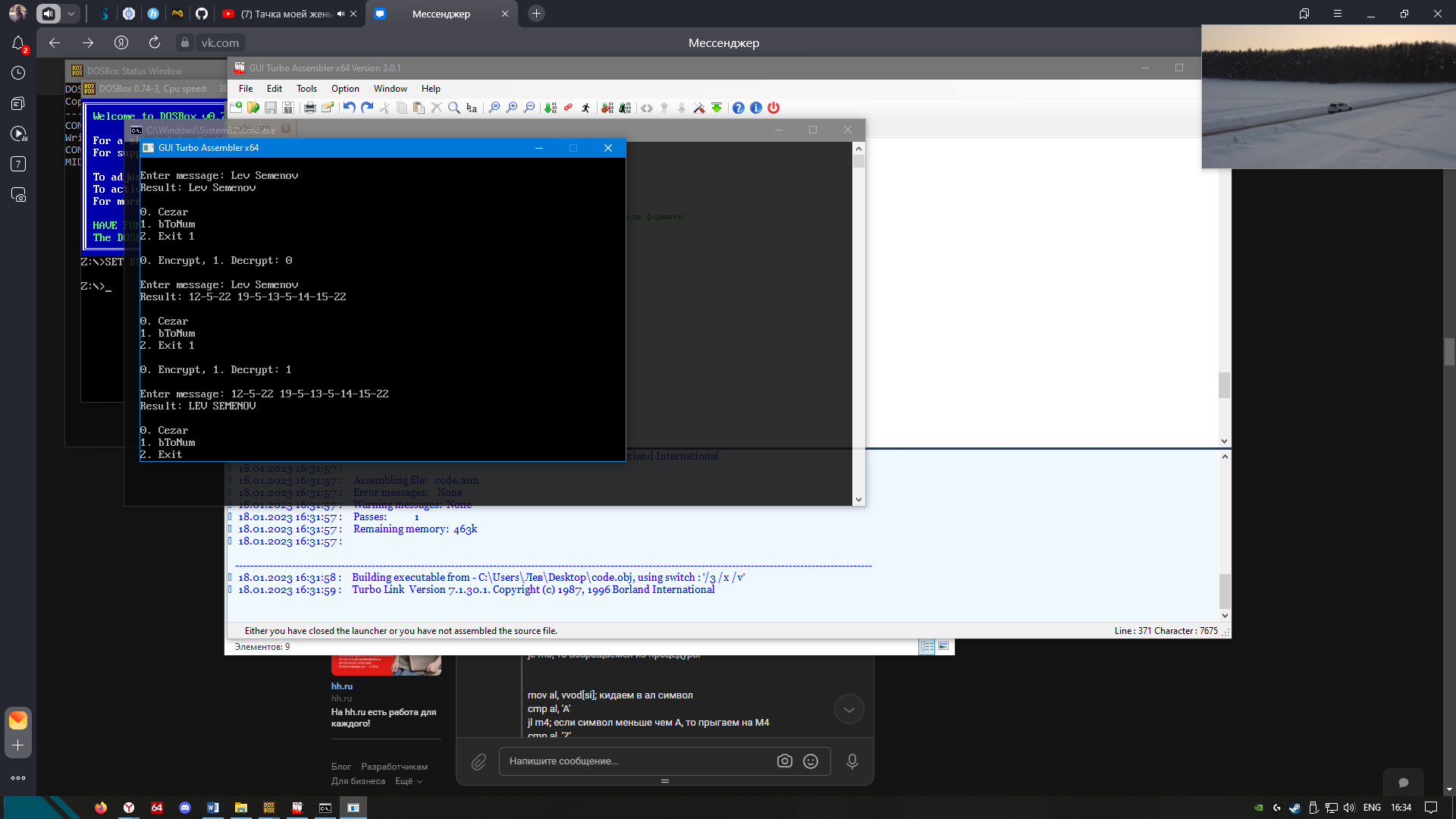


Рис.3.3 Показан результат работы программы по шифрованию и дешифрованию шифром bToNum

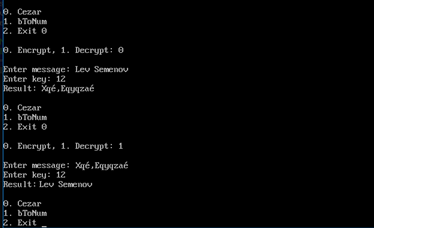


Рис.3.4 Показан результат работы программы по шифрованию и дешифрованию шифром Cezar

## Анализ полученных результатов:

В ходе тестирования программного продукта было выявлено, что поставленная цель создание программного продукта, реализующего два метода шифрования - и все задачи курсовой работы были выполнены.

В результате тестирования были выявлены некоторые неточности в расшифровке текста, которые требуют доработки. Но в целом программа работает с общепринятыми стандартами о шифровании текста.

# Заключение

В процессе написания курсовой работы на тему «создание программного продукта реализующего различные методы шифрования текста» поставленная цель была достигнута.

При достижения поставленной цели были решены следующие задачи:

. Изучены аналогичные программные продукты,

. Изучены теоретические сведения, требуемые для решения поставленной задачи,

. Систематизированы и обобщены полученные знания,

. Создан программный продукт, реализующий три метод шифрования.

При разработке программного продукта были усвоены функции, преобразующие информацию из численного типа в строковый и наоборот, так же были о усвоены условные операторы if…then..else и for. … to…do.

В ходе тестирования программного продукта были выявлены следующие недостатки:

. В ходе работы программы обнаруживаются ошибки.

. В программе не совсем удобное для восприятия меню.

Подводя итоги можно сказать, что при помощи этого программного продукта любой человек, в любом возрасте может шифровать и дешифровать информацию.

# Список литературы

1) Основы микропрограммирования: учебное пособие / О. С. Нургаянова; Уфимск. гос. авиац. техн. ун-т. – Уфа : РИК УГАТУ, 2019. – 113 с.

2) ItSphera [Электронный ресурс]:статья «Криптографические методы и средства защиты информации» © 2010-2014 **IT Sphera**. Внедрение, сопровождение, обновление 1С в Курске.

3) Клод Шеннон - американский инженер и математик «Математическая теория связи». 1945 г.

4) Анин Б. «Защита компьютерной информации»Санкт-Петербург.2000384 с.

5) Байтурин В.Б., Бровкова М.Б., Пластун И.Л. «Введение в защиту информации»Москва ФорумИНФРАМ2000г.107 с.

6) Х.К.А.ван Тилборг Основы Криптологии. Профессиональное руководство и интерактивный учебник. - М., Мир, 2006. - 471 стр.

7) Ростовцев А.Г., Маховенко Е.Б. Теоретическая криптография. - М., Изд. Профессионал, 2005. - 490 стр.

8) В.А. Мухачев, В.А. Хорошко Методы практической криптографии. - М., Изд. Полиграф-Консалтинг, 2005. - 209 стр.

9) Брюс Шнайер, Прикладная криптография. Протоколы, алгоритмы, исходные тексты на языке Си, неофициальный перевод 2-ого издания 1996 г.в.

10) Панасенко С. Алгоритмы шифрования. Специальный справочник. - СПб: БХВ-Петербург, 2009 г., 576 с.

11) Баричев С. Криптография без секретов. - 2004. - 43 стр.

12) Баричев С.Г., Серов Р.Е. Основы современной криптографии. - Горячая Линия - Телеком, 2002 - 153 с.