

MINISTRY OF SCIENCE AND HIGHER EDUCATION OF THE RUSSIAN FEDERATION
FEDERAL STATE AUTONOMOUS EDUCATIONAL INSTITUTION OF HIGHER EDUCATION
"NORTH CAUCASIAN FEDERAL UNIVERSITY"

METHODOLOGICAL INSTRUCTIONS
By implementation of practical works
By discipline
Personal Data Protection in AI Systems

For students directions preparation
11.03.02 "Infocommunication technologies and communication systems" Profile
"Intellectual processing data V
infocommunication systems and networks"

Stavropol, 2026

Content

Introduction

Theoretical information

Practical lesson No. 1. Standard symmetrical encryption **Practical lesson**

No. 2 Research of various methods of protecting text information and their durability based on

key selection

Practical lesson # 3 . Generation simple numbers, used in asymmetric encryption systems

Practical Lesson #4 . Electronic Digital Signature

Introduction

Currently, attention to the problem of information security has sharply increased throughout the world. This is due to the processes of rapid expansion of information flows permeating all spheres of society.

Information has long ceased to be simply a necessary auxiliary resource for production or a byproduct of any activity. It has acquired a tangible value, which is clearly determined by the actual profit derived from its use or the extent of the damage inflicted, with varying degrees of probability, on the information owner. However, the creation of an information processing industry gives rise to a number of complex problems. One such problem is the reliable security and established status of information circulating and processed in information and communications systems and networks.

IN these guidelines have been developed laboratory work aimed at research principles of construction and parameters main units that provide information protection.

1. THEORETICAL INTELLIGENCE

1.1. Problems protection information

Currently, attention to the problem of information security has sharply increased throughout the world. This is due to the processes of rapid expansion of information flows permeating all spheres of society.

Information has long ceased to be simply a necessary auxiliary resource for production or a byproduct of any activity. It has acquired a tangible value, which is clearly determined by the actual profit derived from its use or the extent of the damage inflicted, with varying degrees of probability, on the information owner. However, the creation of an information processing industry gives rise to a number of complex problems. One such problem is the reliable security and established status of information circulating and processed in information and computing systems and networks.

The advent of global computer networks has made it easier for both individual users and large organizations to access information. However, the ease and speed of accessing data via computer networks such as *the Internet* has also made the following data security threats significant if they are not properly protected:

- unauthorized access To information;
- unauthorized change information
- unauthorized access To networks And services;
- other network attacks, For example, repeat intercepted earlier transactions and denial of service attacks.

When processing any significant information using a separate computer, A those more V networks, arises question O her protection from

unauthorized access and use. The most common method of protection in computer systems—the use of passwords—is more suitable for protecting access to computing resources than for protecting information. It is a kind of screen that separates legitimate users of the system from outsiders, through which a qualified user gains access to virtually all information.

Currently, issues related to the storage and transmission of confidential information have acquired exceptional significance in various fields. The problems arising in this regard are addressed *by cryptography* —the science of methods for transforming information to protect it from unauthorized users.

A retrospective look at the history of cryptography's development as a specific area of human activity allows us to identify three main periods. The first, the longest, is the era of "manual cryptography." Its beginning is lost in ancient times, and its end falls in the 1930s. centuries. Cryptography has evolved from a magical art to a completely prosaic applied specialty of officials in diplomatic and military departments.

The second period was the creation and widespread implementation of first mechanical, then electromechanical and electronic encryption devices, and the organization of entire secure communications networks. Its beginning can be considered the development by Gilbert Vernam *in* 1917 of a telegraph encryption machine circuit using *a one-time gamma* (Fig. 1.1).

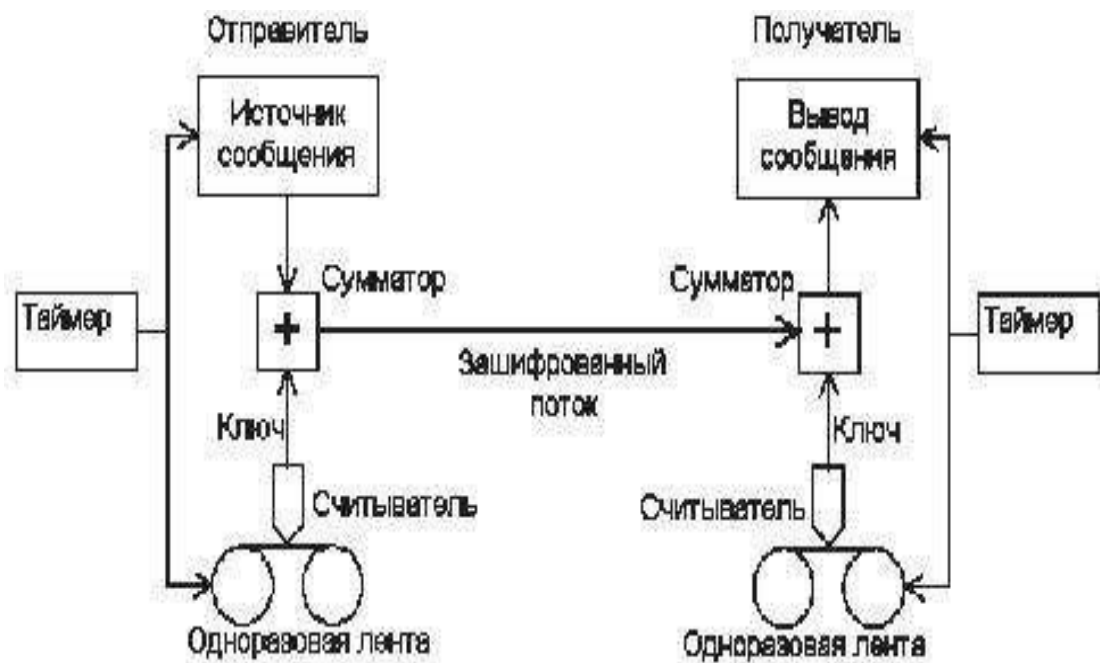


Рис. 1.1. Encryption method Vernama

By the mid-1970s, with the development of extensive commercial communication networks, e-mail and global information systems, new problems came to the fore - problems of key supply and problems of authentication.

In 1976, American scientists Whitfield Diffie *and* Martin Hellman *proposed* two new principles for organizing secure communications without first providing subscribers with secret encryption keys—the principle of so-called *open encryption* and the principle of *open key distribution* . This moment can be considered the beginning of a new period in the development of cryptography. Currently, this area of modern cryptography is developing very intensively .

1.2. From stories cryptography

Concept "safety" covers wide circle interests, both individuals and entire states. In all historical times

essential attention was given problem informational safety, provision protection confidential information from sign-

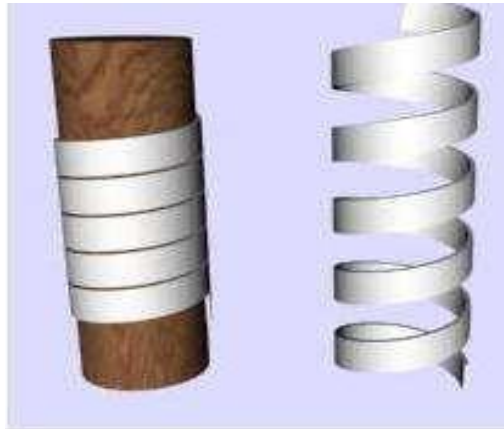
Cryptography deals with the solution of these issues .



Rice. 1.2. John Wallis

Cryptography is secret writing. The term was coined by John Wallis (1616-1703), an English mathematician. The need to encrypt and transmit encrypted messages arose a long time ago. As early as the 5th and 6th centuries BC, the Greeks used a special encryption device. According to Plutarch, it consisted of two cylindrical rods of equal length. and thickness. One was kept for oneself, and the other was given to the departing person. These rods were called *scytals*.

When a message needed to be conveyed, a long strip of papyrus was wound around a scythala, leaving no gaps. Then, leaving the papyrus on the scythala, they wrote whatever was needed on it. Once written, they removed the papyrus and sent it to the recipient without the stem. Since the letters were scattered in disarray, only someone with a scythala of the same length and thickness, and who had wound the papyrus around it, could read the message.



Rice 1.3. Scythala

Square Polybius

In Ancient Greece (2nd century BC) a cipher called the “*Polybius square*” was known.

This device consisted of a 5x5 square, the columns and rows of which were numbered from 1 to 5. One letter was written in each cell (in the Greek version, one cell was empty, and in the Latin version, two letters, *I* and *J*, were placed in one cell). As a result, each letter corresponded to a pair of numbers corresponding to the row and column number.

1	2	3	4	5	
A	B	C	D	E	1
F	G	H	I,J	K	2
L	M	N	O	P	3
Q	R	S	T	U	4
V	W	X	Y	Z	5

13	34	22	24	44	34	15	42	22	34	43	45	32
----	----	----	----	----	----	----	----	----	----	----	----	----

Cogito ergo sum - лат. “Я мыслю, следовательно, существую”
Р.Декарт

Code Caesar

In the 1st century AD, during the war with the Gauls, Julius Caesar, while corresponding with his friends in Rome, replaced the first letter of the Latin alphabet (*A*) with the fourth (*D*), the second (*B*) with the fifth (*E*), and finally the last with the third.

¹ Polybius (200-120 gg. to (AD) ancient Greek historian



ABCDEFGHIJKLMNOPQRSTUVWXYZ
 DEFGHIJKLMNOPQRSTUVWXYZABC

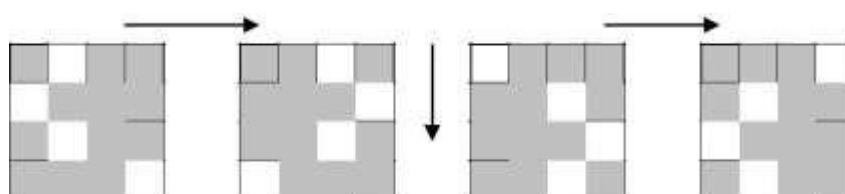
YHQL YLGL YLFL
 Veni vidi vici - "Пришел, увидел
 победил"
 Ю. Цезарь Донесение Сенату о победе
 над понтийским царем

□

The Caesar cipher belongs to the so-called class of *monoalphabetic substitutions* and has many modifications.

Lattice Cardano

Ciphers belonging to the *permutation class* are widely known, in particular the “*Cardano grille*”². This is a rectangular card with holes, most often square, which, when placed on a sheet of paper, leaves only some of its parts open. The number of rows and columns on the card is even. The card is made so that when it is successively turned, each cell of the sheet lying underneath will be occupied. The card is first turned along the vertical axis of symmetry by 180°
 □,
 and then along the horizontal axis also by 180°. And again they repeat the same procedure.

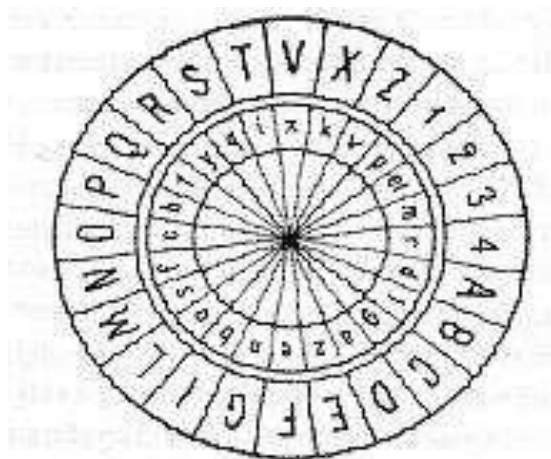


Rice. 1.4. Cardano lattice

² Cardano Gerolamo (1501-1576) - Italian mathematician, philosopher and physician

Disk Alberti

The Italian Alberti (16th century) first put forward the idea of "double encryption" - the text obtained as a result of the first encryption was subjected to re-encryption. In his treatise, Alberti presented his own cipher, which he called "a cipher worthy of kings." He claimed that this cipher was undecipherable. The implementation of the cipher was carried out with the help of a cipher disk, which marked the beginning of a whole series of *polyalphabetic substitutions* . The device consisted of a pair of disks - an outer, fixed one (letters in natural order and numbers from 1 to 4 were applied to it) and an inner, movable one - letters were applied to it. were rearranged. Process encryption concluded V Finding a plaintext letter on the external disk and replacing it with the corresponding ciphertext letter (located beneath it). After encrypting several words, the internal disk was moved one step. The key to this cipher was the order of the letters on the internal disk and its initial position relative to the external disk.



Rice. 1.5. Disc Albert

Table Vigenère

The inconvenience of the monoalphabetic substitution fonts discussed above is obvious, since in the case of using a standard alphabet, the table of frequencies of letters of the alphabet allows one to determine one or several symbols, and this is sometimes enough to break the cipher (Conan Doyle's "The Dancing Men" or Edgar Allan Poe's "The Gold Bug"). Therefore, various techniques were used to complicate deciphering, for example, the use of the "*Vigenere table*", which is a square table with a number of rows and columns equal to quantity letters alphabet. To encrypt any The message is chosen as a slogan word (for example, "monastery") and written above the message with the necessary repetition.

To obtain the ciphertext, find the next character of the slogan, starting with the first in the vertical alphabet, and then the corresponding character of the message in the horizontal alphabet. At the intersection of the selected column and row, we find the first letter of the cipher. Clearly, the key to such a cipher is the slogan used.

m O n A With You r ь m O n A With T

y r ь m o n r a s k And n moose m o r e

sh And r OK O

э о я к щ а п ы й ю й щ о в ч ф ш л ь ш ы

*Table Vigenere*³

ABVGDEZZIYKL M
NOPRSTUFHCCHSHSHCHYEYUYA
BVGDEZZIYKLMNOPRSTUFHCCSHSHCHYEYUYA
A
VGDEZHZIYKLMNOPRSTUFHCCSHSHSHCHYEYU
YAAB
GDEJZIYKLMCIYKLMCIYFYUCHYUCHYUCHYU
YA
DEJZIYKLMNOPRSTUFHCCSHSHSHYEYUYAFBV
G
EZZIYKLMNOPRSTUFHCCSHSHSHYEYUYAFBVG
D
JZIYKLMNOPRSTUFHCCHSHSHSHCHYEYUYAFB
WGDE ZIYKLMNOPRSTUCHSHCHYUYAFBWGDE
IYKLMNOPRSTUFHCCSHSHCHYEYUYAABWGDE
JZ
YKLMNOPRSTUFHCCHSHSHSHYEYUYAABVGDEJZI
KLMNOPRSTUFHCCHSHSHSHYEYUYAABVGDEZ
ZIY
LMNOPRSTUFHTSTUFXYUSHCHYUYABVGDEZH
ZI
MNOPRSTUFHCCHSHSHCHYEYUYAABVGDEZHZ
IYKL
NOPRSTUFHCCSHSHSHCHYEYUYAABVGDEZZIY
KLM
OPRSTUFHCCHSHSHSHYEYUYAABVGDEZHZIYK
LMN PRSTUFHCCYUSHCHYKLMN R
STUFHCCHSHSHY E YUYAABWGDEJZIYKLMNOP
STUFHCCHSHSHSHYEYUYAABVGDEZHZIYKLMN
OPR
TUFHCCSHSHSHYEYUYAABVGDEZZIYKLMNOPR
S
UFHCCHSHSHSHYEYUYAABVGDEZHZIYKLMNOPR

ST

FHCCSHSHSHYEYUYAABWGDZHZIYKLMNOPR

STU

HCCSHSHSHYEYUYAABVGDEZZIYKLMNOPRSTU

F CCHSHCHYUYABVGDEZHZHNOPRSTU

CHSHSHYEYUYAABVGDEZZIYKLMNOPRSTUFHC

SHSHYEYUYAABVGDEZZIYKLMNOPRSTUFHCCH

Sh'EYUYAABVGDEJZIYKLMNOPRSTTUFHCCSH

HIEYUYAABVGZHZHIKSHCHDZH.

YEYUYAABVGDEZZIYKLMNOPRSTUFHCCHSHSH

EYUYAABVGDEZZIYKLMNOPRSTUFHCCCHSHY

YUYAABVGDEZZIYKLMNOPRSTUFHCCSHSHYE

YAABVGDEZHZIYKLMNOPRSTUFHCCHSHCHY

³Blaise de Vigenere (1523-1596) - French ambassador to Rome, wrote a large work O ciphers. Square cipher Vigenère on throughout almost remained undecipherable for 400 years and was considered an undecipherable cipher.

Disposable encryption notebook

An example of an unbreakable cipher is the so-called *one-time pad*, a cipher based on the same idea as the Caesar cipher. Let us call *the* set of letters of the alphabet and punctuation marks $\{ . , : ; ! ? () - " <space> \}$ the extended alphabet; the number of symbols of the extended Cyrillic alphabet in this version will be 44. Let us number the symbols of the extended alphabet with numbers from 0 to 43. Then any transmitted text can be considered as a sequence $\{a_n\}$ of numbers from the set $A = \{0, 1, 2, \dots, 43\}$.

Let us assume that we have a random sequence $\{c_n\}$ of numbers from set A of the same length as the transmitted text – *the key*.

the number from the transmitted text a_n modulo 44 with the corresponding number from the key set c_n :

$$a_n + c_n \equiv b_n \pmod{44}, 0 \leq b_n \leq 43$$

we obtain a sequence of $\{b_n\}$ characters of the ciphertext. To obtain the transmitted text, we can use the same key:

$$a_n \equiv b_n - c_n \pmod{44}, 0 \leq a_n \leq 43$$

Two subscribers engaged in secret correspondence have two identical notebooks. In each of them, a random message is printed on several sheets. subsequence numbers of set A . The sender encrypts their text using the first page of the notebook, as described above. After encrypting the message, they destroy the page they used and send the text to the second subscriber. The recipient of the encrypted text decrypts it and also destroys it.

used notebook sheet. It is easy to see that a one-time cipher is in principle unbreakable, since a symbol in the text can be replaced by any other symbol, and this choice is completely random.

1.3. Encryption methods 1.3.1.

Monoalphabetic method

This method is perhaps the oldest of all known methods. It is based on a simple encryption method: the sender and receiver the body of the encrypted document agree in advance on a certain offset letters relatively their ordinary locations V alphabet. For example, for the Cyrillic alphabet, if the offset is 1, then “А” corresponds letter "Б", "Б" - "В", And So further, A When alphabet fits To the end, then they begin take letters from beginning list. AND it turns out, For example, next: from the word “CODING” we get “LPEYSPGBOYZH”.

A special case of this method is the Caesar cipher discussed earlier. Obviously, any cipher from the class of monoalphabetic methods is not a Caesar cipher (if the alphabetic cardinality of the text is n , then the number of Caesar ciphers is n , and the number of all monoalphabetic ciphers is $n!$). However, even for such methods, it is easy to propose decryption methods based on the statistical properties of the ciphertexts, since the open and closed texts have the same statistical characteristics.

В лабораторной работе № 1 рассматриваются два варианта одноалфавитного метода с фиксированным смещением и с произвольным (задаваемым) смещением.

1.3.2. Encryption method permutations symbols

The essence of this method is that the symbols of the text are rearranged according to certain rules, while only the symbols of the original (unencrypted) text are used.

Permutations in classical cryptography are usually obtained by writing the plaintext and reading the ciphertext along different paths of a geometric figure.

The simplest example The permutation is the recording of the original text by the rows of a certain matrix and reading it by the columns of this matrix .

The sequence of filling the rows and reading the columns can be any and is asked key. Like this in this way, for the matrix size 8×8 (block length 64 characters) there are 1.6×10^9 keys possible, which allows modern computers to decrypt the given text by brute force. However, For matrices size 16×16 (length block 256 characters) there is

1.4×10^{26} keys, and searching through them using modern computing tools is very difficult.

An example of the application of the permutation method can also be an eight-element table with a set of routes called Hamilton routes. The sequence of filling the table each time corresponds to the numbering of its elements. If the length of the encrypted text is not a multiple of the number of elements, then during the last filling, an arbitrary symbol is entered into the free elements. Selection from the table for each filling can be carried out in one's own way route, and the routes can be used either sequentially or in the order specified by the key.

Permutation methods are characterized by the simplicity of the algorithm, the possibility of software implementation and a low level of protection, since with a large length of the original text, statistical patterns of the key appear in its encrypted version, which allows it to be quickly revealed. Another flaw these methods - lung disclosure, If uda-

It is necessary to send several specially selected messages to the system for encryption. Thus, if the length of the block of the original text is equal to K symbols, then to reveal the key it is sufficient to pass $K-1$ blocks of the original text through the encryption system, in which all the symbols, except one, are the same.

1.3.3. Encryption inverse symbols (By addition to 255)

This method encryption, is private by chance monoalphabetic twisted substitution in the alphabet of cardinality 256. The essence of the method consists of replacing the *ASCII character* with the number i with the character with the number $255-i$. The decryption operation is carried out similarly.

1.3.4. Polyalphabetic methods encryption

Polyalphabetic encryption (polyalphabetic substitution) consists of using monoalphabetic methods with different keys for successive characters of the encrypted text.

For example, the first symbol is replaced using the Caesar method with an offset of 14, the second with an offset of 10, and so on until the end of the given key. The procedure then continues periodically. A more common situation is when not the Caesar cipher is used, but a sequence of arbitrary substitutions corresponding to monoalphabetic methods.

A more illustrative example of this type of encryption is the gamma method. This method of transformation consists of symbols closed text sequentially add up With symbols of a special sequence called a gamma. This transformation is sometimes called gamma mapping onto plaintext.

Actually procedure overlays Maybe be carried out one in two ways:

1) The symbols of the closed text and the gamma are replaced by digital equivalents and then added modulo K , where K is the number of alphabet symbols,

$T_{sh} = (T_o \oplus T_g) \bmod K$ Where T_{sh} - ciphertext,
 T_o - plain text,
 T_g - gamma.

2) Text and gamma symbols are represented in binary codes, and then each pair of binary digits is added modulo 2.

The strength of gamma encryption is determined primarily by the quality of the gamma, which is determined by two characteristics: the length of the period and the randomness of the distribution over the period.

The length of a gamma period is the minimum number of symbols after which the sequence begins to repeat. The random distribution of symbols over a period means the absence of patterns between the appearance of different symbols within a period.

В лабораторной работе № 1 рассматриваются три варианта многоалфавитного метода – с фиксированным ключом, с ключом фиксированной длины и с ключом произвольной длины.

1.3.5. Basic requirements for information encryption methods

1) The complexity and labor intensity of encryption and decryption procedures should be determined depending on the level of secrecy of the data being protected.

2) The security of the closure must be such that secrecy is not violated even if the intruder knows the closure method .

3) The method of closing and the set of service data (key settings) used should not be too complex. The costs of protective transformations should be acceptable for a given level of information security.

4) The execution of the procedures for direct and inverse transformations should be formal and as simple as possible.

5) The procedures for direct and inverse transformation should not depend on the length of the message.

6) Errors occurring during the conversion process should not propagate throughout the system and cause information loss. The occurrence of errors in the transmission of an encrypted message over communication channels should not preclude the possibility of reliably decrypting the text at the receiving end.

7) The message redundancy introduced by closure should be as small as possible.

8) Volume key Not must make it difficult his memorization And forwarding.

1.3.6. Histogram text

One of the most well-known methods of cryptanalysis is the study of the statistical characteristics of ciphertexts. A graphical representation of the total frequency of occurrence of symbols in a text is called a histogram of that text.

Let's assume we're dealing with a monoalphabetic encryption method. Knowing the frequency of letters in the alphabet, we can guess which letter was replaced with a given one. For example, the frequently occurring letter "O" is replaced with the rarely occurring letter "Щ."

Для наглядности, в лабораторной работе № 1 используются двойные гистограммы, отображающие частоту встречаемости символов в исходном и зашифрованном текстах.

It follows have V view, What view histograms For standard distribution depends from species original text next way: If

If the source text contains Cyrillic and Latin alphabet characters, then the statistical distribution is output for Cyrillic and Latin, if only Cyrillic alphabet (Latin) That is being output statistical distribution for Cyrillic (Latin).

Practical lesson No. 1

Usage classical cryptoalgorithms substitutions and permutations to protect text information

The aim of the work: to develop students' practical skills in studying classical cryptographic algorithms of polyalphabetic substitution, polyalphabetic substitution and rearrangement for the protection of text information and formation histograms displaying the frequency of occurrence of symbols in the text for cryptanalysis of classical codes.

1. Description work

L_LUX.EXE program . A window appears on the display screen, with a text editor located in its center (for displaying encrypted and decrypted texts). The main menu is located in the top line of the window, allowing the user to perform the desired action. Just below the main menu is a toolbar (for managing quick command buttons and other "hot" controls). At the very bottom of the window, under the text editor, is a status bar, which indicates a hint and displays additional information. For convenience, the toolbar keys are equipped with pop-up tips.

To access the main menu, press *F10*. Navigate through the main menu using the cursor keys. To access a menu item, press "ENTER." To return to the main menu or exit it entirely, press "ESC."

Now let's take a closer look at each of the main menu items.

1.1. Editor

This main menu item contains the following sub-items: create document, open file, save file, exit program.

Initially, immediately after starting the program, the text editor is unavailable, as are almost all items in the main menu (except for creating a document, opening a file, exiting the program, and information about the program) and most of the control panel keys (except for creating a document, opening a file, and exiting the program).

Create document (Ctrl+N) – this sub-item makes the text editor available for work (the user gets the right to create his own text file, which can later be used when working with the program), It also becomes possible to use all previously inaccessible main menu items and control panel keys.

Open File (Ctrl+L) – when selected, a dialog box appears allowing you to select a file to upload. The file's contents will be displayed in the text editor window.

Similar to the “Create Document” item, a text editor with displayed text becomes available for work, and it also becomes possible to use all previously inaccessible items of the main menu and control panel keys.

Save file (Ctrl+S) - when you select this option, a dialog box appears that allows you to save the contents of the text editor to disk .

Exit the program (Ctrl+X) - when you select this item, a dialog box appears that allows you to exit the program.

1.2. Histogram

Display two histograms on the screen, showing the frequency of occurrence of characters in the text.

***Внимание !** До выполнения шифрования и дешифрования вызывать гистограмму не имеет смысла, так как еще не сформированы тексты, для которых будет просматриваться гистограмма.*

There is opportunity viewing following combinations histograms:

- histograms original And encrypted file,
- histograms encrypted And decrypted file,
- histograms standard distribution And encrypted text-hundred,
- histograms of the standard distribution and the deciphered one text.

In histograms, the left and right mouse buttons are used to zoom.

For example, after encrypting a large amount of text, the user wants to view the histograms of the original and encrypted files. Since the text size is quite large, two histograms with a large number of columns in each will be displayed on the screen (a column corresponds to one character of the text), but it will be difficult to tell which one of these columns corresponds to a particular character of the text, and what is the frequency of occurrence of this character. Therefore, the user has the opportunity to increase the scale of any of the two histograms in order to more accurately determine the required value of the frequency of occurrence of a specific character. To do this, you need to move the mouse pointer to the left border of the section of the histogram that you want to enlarge, then press the left mouse button and, without releasing it, stretch the rectangle so that its lower right corner coincides with the right border of the enlarged plot histograms. After this follows let go left click the mouse and an enlarged image of the desired area will appear on the screen.

Moreover, by pressing and holding the right mouse button, you can move the histogram in any direction in order to study the entire resulting distribution on a larger scale.

To return to the original view from the enlarged scale, you need to move the mouse pointer over the histogram, then press the left mouse button and, without releasing it, drag a small rectangle from the bottom up. After this, you should release the left mouse button and the original image of the histogram will appear on the screen.

1.3. Encryption

Execution encryption text file is being carried out one of the seven methods discussed in the laboratory work.

1. Monoalphabetic method (With fixed displacement).
2. Monoalphabetic method With given displacement (from 2 up to 20).
3. Permutation symbols.
4. By addition to 255 (inverse method).
5. Polyalphabetic method (With fixed key).
6. Polyalphabetic method With key fixed lengths.
7. Polyalphabetic method With key arbitrary lengths.

The choice of encryption method is made using the mouse, the cursor keys and the “ENTER” key.

A window then appears in which, depending on the encryption method, you must specify various parameters and either confirm or cancel the encoding process. The encrypted text will then be displayed in the editor window.

1.4. Decoding

Similar to the previous point, the decryption method is selected (it must match the method by which the file was encrypted).

A window appears again in which, depending on the decryption method, you must specify certain parameters and either confirm the decryption process or refuse it.

After this, the decrypted text will be displayed in the editor window. If decrypted correctly, the resulting text will match the original .

1.5. Additional information

The program provides the opportunity to view additional information ('Help Ctrl+F9'), reference information about the encryption methods used ('About methods Ctrl+F10'), information about the program ('About the program Ctrl+F11').

2. Formed competencies

This laboratory work is aimed at developing the following competencies of a bachelor's degree graduate:

OPK-1: ability install software And hardware for information and automated systems;

PC-5: the ability to interface hardware and software as part of information and automated systems.

3. Theoretical Part

To complete the lab, you need to run the L_LUX program. A window with a text editor located in the center will appear on the display screen. The main menu is located in the top line of the window, allowing the user to perform the required action. Just below the main the menu contains a toolbar, and at the very bottom of the window, under the text editor, there is a status bar, which indicates a hint and is being output additional information. Cl-

For your convenience, your toolbars are equipped with pop-up tips .

To access the main menu, press the F10 key. Navigate through the main menu using the mouse.

Example work With program.

Let's consider monoalphabetic encryption With fixed key. Download V window editor original text (any) length from 10 to 15 sentences. Save it; you'll need to do this for future work with this file. Then, select the ENCRYPTION menu item and select the monoalphabetic method (with a fixed offset). In the window that appears, press the ENCRYPTION button. After encryption is complete, you can view the ciphertext in the editor.

Go to the HISTOGRAM menu item. Select the histogram type that displays the histograms of the original and ciphertext. Analyze the histograms. They should look roughly the same .

To find out the cipher key (the offset of the second alphabet relative to the first), it is necessary to find symbols that have the same value using the histograms. frequency occurrence. For example, the most common symbol V The character in the first histogram must be converted to the most frequent character in the second histogram during encryption. Thus, by finding the two most frequently occurring characters in both histograms, the offset can be calculated using the standard ASCII code table. Knowing the offset and the character encoding table, the text can be easily decrypted. By calling the DECRYPTION menu, the same actions can be performed automatically.

Note. When encrypting and decrypting from the coding table, symbols with codes 176-223 and 240-255 are not used, that is, when decrypting manually, these symbols should be skipped and it should be assumed that, for example, the symbol Я has code 159, and not 223, similarly II not 175, but 239.

Sometimes in histograms, under the columns showing the frequency of occurring symbols, not the symbols themselves are shown, but their table codes in square brackets.

4. Equipment And materials

The hardware for laboratory work includes computer classrooms with installed network equipment, as well as

- operating room system Windows;
- program L_LUX V environment C++.

5. Instructions By technology security

Requirements security before beginning works:

- PROHIBITED: changing clothes, using fire, smoking, eating in the laboratory;
- Ensure that electrical outlets and connectors are intact. Do not operate, break, or attempt to repair faulty power tools.
- V laboratories necessary be V replaceable shoes;
- Turn on the computer only after receiving permission for the work being performed and the permission of the teacher or laboratory assistant.

Requirements security in time works:

- When performing laboratory work, students are required to use only computers, peripheral equipment, connecting cables, measuring equipment and diskettes that are directly related to the given laboratory lesson;
- connection and disconnection of components of the computing complex should be carried out only with complete removal of tension from all components computing complex.

Requirements security By at the end works:

- at the teacher's command, students are required to turn off the computer and components of the computing complex;
- working place bring V order.

6. Tasks

Task No. 1. For the monoalphabetic method with a fixed offset, determine the offset set in the program. To do this, you should:

- view your text file previously created using the editor;
 - execute For this file encryption;
 - view V editor encrypted file;
 - view histograms original And encrypted texts;
 - describe the histograms (how they are similar, how they are different)
- and determine with what offset the encryption was performed;
- decipher encrypted text:
 - Using a histogram, then check the correctness of the decryption in the editor;
 - Manually, using histograms, describe and explain the decryption process.

The report for each encryption method describes the sequence of actions performed, the names of all files used, the resulting histograms, the offset found, and the decryption process.

The teacher is provided with a report on the work completed and all files used and created.

Task #2. For the monoalphabetic method with a given offset (Caesar cipher), the following should be done:

- perform encryption with arbitrary offset on your plaintext;
- view And describe histograms original And encrypted

texts, define bias For several symbols;

- decipher text With help programs;
- Decrypt the text encrypted with the Caesar cipher using the program using the offset search method; indicate in the report at what offset the file was encrypted.

Task No. 3. For the symbol permutation method, decrypt the encrypted file. For this necessary define law Permutations of plaintext characters. Create a small file, a few characters long, containing known plaintext, and encrypt it. Look at the histograms, describe them, and determine whether useful decryption information can be extracted from them. Compare your plaintext and the ciphertext using an editor and determine the law of permutations of characters.

Decipher file:

- manually And explain yours actions;
- With help programs.

Exercise No. 4 . For inverse coding (By addition to 255)

- complete encryption For his own arbitrary file;
- look at the histograms of the original and ciphertexts, describe the histograms and determine the offset for several characters;
- Decrypt the encrypted text, check the correctness of the decryption in the editor.

Task #5 . For polyalphabetic encryption with a fixed key determine, How many monoalphabetic methods And with what The offset is used in the program. To do this, you need to create your own file consisting of a string of identical characters, encrypt it, and use the histogram to determine the encryption method and the set of offsets.

Task No. 6. For polyalphabetic encryption with a fixed-length key:

- perform encryption and determine from the histogram what offset each symbol receives for a file consisting of a string of the same

out symbols;

- perform encryption and decryption for file of arbitrary text;
- Review and describe the histograms of the original and ciphertexts;

answer what information can be obtained from the histograms.

Exercise No. 7 . For polyalphabetic encryption With with a random password, the task is completely similar to task 6.

7. Content report

Report must contain:

1. Name work;
2. target work;
3. workers schemes;
4. main ratios And settlement formulas;
5. tables;
6. conclusions By work.

8. Control questions

Provide answers to the test questions in the report in accordance with the option number specified by the teacher (Table 1.1).

Table 1.1 - Control questions

Number options	Control questions
1, 5, 7, 3, 9, 18, 28	Which You you know methods cryptographic protection files?
2, 4, 6, 8, 20, 22, 24, 26, 30	IN how advantages And flaws monoalphabetic me- Todov?
11, 13, 15, 10, 17, 19, 27	If necessary encrypt text, containing important- nuyu information, Which method from considered You will you choose? Justify mine choice
12, 14, 16, 21, 23, 25, 29	It is advisable whether again apply For already encrypted- roved text:

	A) method polyalphabetic encryption; b) method Caesar?
--	---

9. A list of recommended literature on this topic

1. Zhuk, A.P., Zhuk, E.P., Lepeshkin, O.M., Timoshkin, A.I. Information Security: A Textbook for Bachelors [Text] / A.P. Zhuk et al. - M.: INFRA-M, 2015. -392 p.

2. Shan'gin, V.F. Computer Information Protection. Effective Methods and Tools [Text] / V.F. Shan'gin – M.: DMK Press, 2012. – 544 p.

Practical lesson No. 2

Study various methods protection text information and their durability based on key selection

The aim of the work: to study encryption/decryption methods by symbol permutation, substitution, gamification, using the Vigenere table, and to compare the strength of various methods based on attacks by trying all possible keys.

This lab examines a method for cracking a cipher based on trying all possible key variants. The criterion for a correct variant is availability V text "probable words".

Moves over set everyone possible keys, encrypted text is decrypted on each key. In the resulting "pseudo-plaintext"

A probable word is searched for. If such a word does not exist, the current text is rejected and the next key is selected. If such a word is found, a key variant is displayed on the screen. The search for keys then continues until the entire set of variants is exhausted. It is possible to find several keys for which the "pseudo-plaintexts" contain a probable word.

After completion enumeration necessary decipher text on found keys.

"Pseudo-open text" is being output on screen For visual control.

If the operator recognizes the text as clear, the cracking process is complete. Otherwise, the current key variant is rejected and the next key is selected.

1. Formed competencies

This laboratory work is aimed at developing the

following competencies of a bachelor's degree graduate:

OPK-1: ability install software And hardware for information and automated systems;

PC-5: the ability to interface hardware and software as part of information and automated systems.

2. Theoretical Part

LAB_RAB.exe program , which is used for encryption/decryption, as well as decryption (using the probable word pulling method) of files. The system implements the following functions:

- input, removal And selection keys user;
- support list keys;
- encryption And decryption text;
- decryption text by way selection keys, method probable word

stretches.

The system supports the following methods of cryptographic transformation of information:

- replacement;
- permutation;
- gamming;
- table Vigenère.

When running encryption and decryption utilities, the user is asked to confirm the correctness of the selected method for work and the compliance of the specified key with the user's purposes (also, whenever changing a file in a text editor, a request is issued to save changes at each step, further development of which will lead to irreversible changes in the file and the loss of the original information).

Description interface.

- Window text editor With wide set additional functions.
- Table of all keys entered into the system with quick access to enter, delete or select the current key.
- List of all encryption methods for quick and easy switching between them.

- The main thing menu (up screen).
- Additional menu (called by pressing right buttons We-shi).
- Kit auxiliary buttons For fast And convenient inter-face.
- Fields conclusion current states systems:
- current key;
- probable word;
- strength key For broaches.

Example work With program

***Внимание!** Будьте внимательны при установке параметров работы, так как в процессе вычисления по ходу работы эти параметры изменить уже не удастся. После запуска программы абсолютно все рабочие поля пустые и необходимо провести первоначальные настройки для работоспособности системы.*

1. Introduced list keys.
2. Introduced probable word (optional) at first, to his (After entering all the menus for starting the drawing, they are still unavailable).
3. It is selected necessary method encryption.
4. Loading original or encrypted file (the corresponding menus for encryption and decryption open).
5. It's starting up necessary process:
 - encryption;
 - decryption;
 - broach probable words;
 - conversion text.
6. Continuation work V any in order V in accordance With described points.

7. When you finish working, do not forget to save the necessary results (when closing and loading new files, the system automatically asks for confirmation to save).

Encryption

1. Open file.
2. Contribute necessary changes.
3. Tune corresponding parameters:
 - type encryption;
 - key;
 - others.
4. Launch process encryption through paragraph menu UTILITIES \ ENCRYPT F5.

Внимание! При шифровании файла все внесенные пользователем изменения до текущего момента времени будут сохранены на жестком диске.

Decoding

1. Open file.
2. Produce necessary changes.
3. Tune corresponding parameters:
 - type encryption;
 - others.
4. Launch process decryption through paragraph menu UTILITIES \ DECIPHER F6.

Внимание! При расшифровании файла все проведенные пользователем изменения до текущего момента времени будут сохранены на жестком диске.

Broaching probable words (decryption)

Внимание! Мощность ключа задается заранее в опции "сила ключа". Длина ключа значительно влияет на время протяжки вероятного слова (в худшем случае имеем дело с логарифмическим алгоритмом).

1. Introduced probable word (long from 1(3) to 9)
2. For departments again found keys from previous the inscription "selection" is added between them.

3. Overkill keys.
4. It stands for first all line text By current key.
5. The contents of this line are compared with the meaning of the probable word in portions equal to the length of the probable word.
6. If found at least one coincidence, we remember key.
7. Let's move on To new key.
9. The results must be contained in the key list. If no matches are found, nothing is added to the key list.

Operations With keys

WITH base keys can be carried out next actions:

- add new key;
- delete one recording;
- change active recording;
- clear all table introduced keys .

<i>Примечание. Под словами "работа с таблицей ключей" имеются ввиду ключи, введенные для использования в двух методах (гаммирования и таблица Виженера).</i>

Keys For permutations

At any given time, there can only be one current key for permutation in the system.

Rules input key for rearrangement:

1. When you switch to the "Permutation" item in the list of encryption methods supported by the system, a window for entering the permutation key is called up. The window consists of two buttons (Cancel and Exit without changes) and the Enter button – confirmation of the set key length) and the window for setting the key length for rearrangement.

2. In the key length setting window, you must select the required length (parameters can be changed within the range 1...9) and confirm your desire to use a key of exactly this length.

3. After confirmation, the window will display buttons with numbers on the front side. (in quantity, equal length key), at pressing the button is fixed (discolored) for

the impossibility of its further use (since all numbers in the permutation key must be unique).

4. After going through all the buttons, the system remembers the entered key, displays it in the key input field and exits the permutation key input window to the main program window.

4. Equipment And materials

The hardware for laboratory work includes computer classrooms with installed network equipment, as well as

- operating room system Windows;
- program *LAB_RAB V* environment C++.

5. Instructions By technology security

Requirements security before beginning works:

- PROHIBITED: changing clothes, using fire, smoking, eating in the laboratory;
- Ensure that electrical outlets and connectors are intact. Do not operate, break, or attempt to repair faulty power tools.
- V laboratories necessary be V replaceable shoes;
- Turn on the computer only after receiving permission for the work being performed and the permission of the teacher or laboratory assistant.

Requirements security in time works:

- When performing laboratory work, students are required to use only computers, peripheral equipment, connecting cables, measuring equipment and diskettes that are directly related to the given laboratory lesson;
- connection And disconnection components computing the complex should be carried out only after the voltage has been completely removed from all components computing complex.

Requirements security By at the end works:

- at the teacher's command , students are required to turn off the computer and components of the computing complex;
- working place bring V order.

6. Tasks

LAB_RAB.exe program , which is used for encryption/decryption, as well as decryption (using the probable word pulling method) of files.

System implements next functions:

- input, removal And selection keys user;
- support list keys;
- encryption And decryption text;
- decryption text by way selection keys, method probable word

stretches.

The system supports the following methods of cryptographic transformation of information:

- replacement;
- permutation;
- gamming;
- table Vigenère.

When running encryption and decryption utilities, the user is asked to confirm the correctness of the selected method for work and the compliance of the specified key with the user's purposes (also, when changing a file in a text editor, a request is issued to save changes at each step, the further development of which will lead to irreversible changes in the file and loss of original information).

Description interface:

1. A text editor window with a wide range of additional functions.

2. Table of all keys entered into the system with quick access to enter, delete or select the current key.

3. List of all encryption methods for quick and easy switching between them.

4. The main thing menu (up screen).

5. Additional menu (called by pressing right buttons We-shi).

6. Kit auxiliary buttons For fast And convenient interface.

7. Fields conclusion current states systems:

- current key;
- probable word;
- wrench force for pulling.

Example work With program

Important! Be careful when setting the operating parameters, as these parameters cannot be changed during the calculation process.

After launching the program, absolutely all working fields are empty and it is necessary to perform initial settings for the system to function properly.

1. Introduced list keys.

2. Introduced probable word (optional) at first, to his (After entering all the menus for starting the drawing, they are still unavailable).

3. It is selected necessary method encryption.

4. Loading original or encrypted file (the corresponding menus for encryption and decryption open).

5. It's starting up necessary process:

- encryption;
- decryption;
- broach probable words;

– conversion text.

6. Continue working in any order in accordance with the points described.

7. When you finish working, do not forget to save the necessary results (when closing and loading new files, the system automatically asks for confirmation to save).

Encryption

1. Open file.

2. Contribute necessary changes.

3. Tune corresponding parameters:

– type encryption;

– key;

– others.

4. Launch process encryption through paragraph menu

UTILITIES \ ENCRYPT F5.

Important! When encrypting a file, all changes made by the user up to the current point in time will be saved to the hard drive.

Decoding

1. Open file.

2. Produce necessary changes.

3. Tune corresponding parameters:

– type encryption;

– others.

4. Launch process decryption through paragraph menu

UTILITIES \ DECIPHER F6.

Attention! When decrypting a file, all changes made by the user up to the current point in time will be saved on the hard drive.

Broaching probable words (decryption)

Attention! The key strength is preset in the "Key Strength" option. The length of the key significantly affects the time it takes to pull the probable va (V the worst case we have case With logarithmic algorithm).

1. Introduced probable word (long from 1(3) to 9)
2. For departments again found keys from previous the inscription "selection" is added between them.
3. Overkill keys.
4. It stands for first all line text By current key.
5. The contents of this line are compared with the meaning of the probable word in portions equal to the length of the probable word.
6. If found at least one coincidence, we remember key.
7. Let's move on To new key.
8. Let's move on To next line.
9. The results must be contained in the key list. If no matches are found, nothing is added to the key list.

Operations With keys

WITH base keys can be carried out next actions:

- add new key;
- delete one recording;
- change active recording;
- clear all table introduced keys.

Note: The words "working with a key table" refer to keys introduced for use in the two methods (gamma and Vigenere table).

Keys For permutations

At any given time, there can only be one current key for permutation in the system.

Rules input key for rearrangement:

1. When you switch to the "Permutation" item in the list of encryption methods supported by the system, a window for entering the permutation key is called up.

installation. The window consists of two buttons (Cancel and Exit without changes) and the Enter button – confirmation of the set key length) and the window for setting the key length for rearrangement.

2. In the key length setting window, you must select the required length (parameters can be changed within the range 1...9) and confirm your desire to use a key of exactly this length.

3. After confirmation, the window will display buttons with numbers on the front side. (in quantity, equal length key), at pressing The button is fixed (discolored) to prevent its further use (since all numbers in the permutation key must be unique).

4. After going through all the buttons, the system remembers the entered key, displays it in the key input field and exits the permutation key input window to the main program window.

Exercise No. 1.

1. Get acquainted With description laboratory work And task.

2. Configure the program: select the encryption method; enter keys for all methods; enter a probable word; perform all other system settings.

Exercise №2. For method replacements (monalphabetic method):

- choose given algorithm V list available methods cipher-

nia;

- install necessary bias;

- open arbitrary file;

- view content original file;

- execute For this file encryption (at necessity

Can ask Name encrypted file);

- view V editor encrypted file;

- enter probable word;

- enter probable length key (except method replacements);

- pick up key;
- execute decryption with by everyone found keys;
- find V any from decrypted files Right decrypted keyword;
- decipher file original key;
- check result.

Exercise №3. For permutation method :

- choose method permutations;
- in the opened window for entering the character permutation key, first indicate the length of this key, and then create the necessary combination for the key from the buttons that appear, pressing the buttons in the specified order; in this case, the buttons already used become unavailable to prevent their re-entry;
- further actions are completely consistent with those described in the previous paragraph of the task.

Exercise No. 4. For gamma method :

- choose gamma method ;
- enter key gamming;
- fully repeat from tasks 3 second paragraph.

Task No. 5. For the Vigenere table, all actions are repeated from task 4 (the gamma method).

The report for each encryption method describes the sequence of actions performed, indicates the names of all files used, the original and found keys, and describes the decryption process .

The teacher is provided with a report on the work done and all files used.

7. Content report

The report describes the sequence for each encryption method. performed actions, are indicated names everyone used-

files, original and found keys, the decryption process is described. ing.

To the teacher provided report O done work And all used files.

8. Control questions

Номер варианта	Контрольные вопросы
1,5,7, 3,9,18,28	Чем отличается “псевдооткрытый” текст (текст, полученный при расшифровке по ложному ключу) от настоящего открытого текста?
2,4,6,8, 20,22,24, 26,30	Как зависит время вскрытия шифра описанным выше способом подбора ключей от длины “вероятного” слова?
11,13,15, 10,17,19, 27	Зависит ли время вскрытия шифра гаммирования (или таблицы Виженера) от мощности алфавита гаммы?
12,14,16 21,23,25, 29	В чем недостатки метода дешифрования с использованием протяжки вероятного слова?

9. A list of recommended literature on this topic

1. Zhuk, A.P., Zhuk, E.P., Lepeshkin, O.M., Timoshkin, A.I. Information Security: A Textbook for Bachelors [Text] / A.P. Zhuk et al. - M.: INFRA-M, 2015. -392 p.

2. Shan'gin, V.F. Computer Information Protection. Effective Methods and Tools [Text] / V.F. Shan'gin – M.: DMK Press, 2012. – 544 p.

Practical lesson No. 3

Standard symmetrical encryption *Aes Rijndael*

The purpose of the work: to study the encryption principles used in the symmetric encryption algorithm *AES RIJNDAEL* and the ability to draw independent conclusions based on research results.

2. Formed competencies

This work is aimed at developing the following competencies of a bachelor's degree graduate:

OPK-3: ability install software And hardware for information and automated systems;

PC-5: the ability to interface hardware and software as part of information and automated systems.

3. Theoretical Part

State, encryption key, and number of rounds. Rijndael is an iterative block cipher with variable block lengths and variable key lengths. The key length and block length can be independently 128, 192 or 256 bits.

Various transformations operate on an intermediate result called State.

The state can be represented as a rectangular array of bytes. This array has 4 rows, and the number of columns is denoted by N_b and is equal to the block length divided by 32.

The encryption key is also represented as a rectangular array with four rows. The number of columns is denoted by N_k and is equal to the key length divided by 32. This is shown in Figure 1.

In some cases, the encryption key is shown as a linear array of four-byte words. The words consist of four bytes, which are located in one column (when represented as a rectangular array).

$a_{0,0}$	$a_{0,1}$	$a_{0,2}$	$a_{0,3}$	$a_{0,4}$	$a_{0,5}$
$a_{1,0}$	$a_{1,1}$	$a_{1,2}$	$a_{1,3}$	$a_{1,4}$	$a_{1,5}$
$a_{2,0}$	$a_{2,1}$	$a_{2,2}$	$a_{2,3}$	$a_{2,4}$	$a_{2,5}$
$a_{3,0}$	$a_{3,1}$	$a_{3,2}$	$a_{3,3}$	$a_{3,4}$	$a_{3,5}$

$k_{0,0}$	$k_{0,1}$	$k_{0,2}$	$k_{0,3}$
$k_{1,0}$	$k_{1,1}$	$k_{1,2}$	$k_{1,3}$
$k_{2,0}$	$k_{2,1}$	$k_{2,2}$	$k_{2,3}$
$k_{3,0}$	$k_{3,1}$	$k_{3,2}$	$k_{3,3}$

Drawing 1. Example performances States (Nb=6) And Key cipher-razvaniya (Nk=4)

The input to the cipher is denoted as state bytes in the order $a_{0,0}$, $a_{1,0}$, $a_{2,0}$, $a_{3,0}$, $a_{0,1}$, $a_{1,1}$, $a_{2,1}$, $a_{3,1}$... After the cipher completes, the output is obtained from the state bytes in the same order. The number of cycles is denoted as N_r and depends on the values of N_b and N_k .

Table 4.1. Number of cycles (N_r) as a function of key length and block length

N_r	Nb = 4	Nb = 6	Nb = 8
$N_k = 4$	10	12	14
$N_k = 6$	12	12	14
$N_k = 8$	14	14	14

Cyclic transformation. Cyclic transformation consists of four different transformations: byte replacement, string shift, mixing columns, addition cyclic key. Last cycle is different from a simple cycle only by the absence of mixing of columns.

```

Round (State, RoundKey)
{
ByteSub(State); // replacing bytes
ShiftRow(State); // shifting rows
MixColumn(State); // kneading columns
AddRoundKey(State, RoundKey); // adding cyclic key
}

```

Last cycle cipher A little differs:

```
FinalRound(State, RoundKey)
{
  ByteSub(State); // replacement byte
  ShiftRow(State); // shift lines
  AddRoundKey(State, RoundKey); // adding cyclic key
}
```

As you can see, the last cycle differs from the simple cycle only in the absence of column shuffling. Each of the transformations presented is discussed in detail below.

Transformation kneading columns (MixColumn)

The transformation is the multiplication of the state by the matrix ME during encryption or the matrix MD during decryption:

$$\begin{array}{l}
 \text{ME} \quad \begin{array}{|c|c|} \hline \square & \square \\ \hline \end{array} \text{State} \\
 \text{MD} \quad \begin{array}{|c|} \hline \square \\ \hline \end{array} \text{State}
 \end{array}$$

$$\begin{array}{|c|c|c|c|} \hline 2 & 3 & 1 & 1 \\ \hline 1 & 2 & 3 & 1 \\ \hline 1 & 1 & 2 & 3 \\ \hline 3 & 1 & 1 & 2 \\ \hline \end{array} \otimes \begin{array}{|c|c|c|c|} \hline b1 & b5 & b9 & b13 \\ \hline b2 & b6 & b10 & b14 \\ \hline b3 & b7 & b11 & b15 \\ \hline b4 & b8 & b12 & b16 \\ \hline \end{array}$$

$$b1 = (b1 * 2) \text{ XOR } (b2 * 3) \text{ XOR } (b3 * 1) \text{ XOR } (b4 * 1)$$

Multiplication two byte is happening By to the next one algorithm:

- If one from byte equal 0, result will 0;
- If one from byte equal 1, result will another byte;
- in other cases, each byte is replaced according to table L, the replaced bytes are added up, if necessary, 255 is subtracted to get into interval [0, 255] and a replacement is made according to table E, which gives the result.

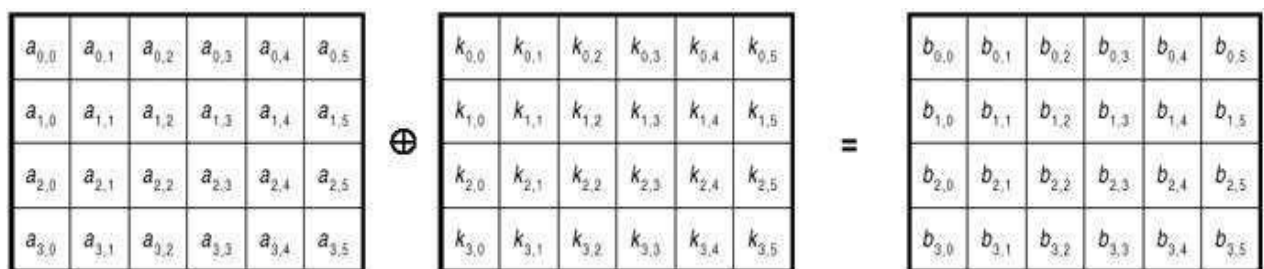
$L = \{$
 , 0x00, 0x19, 0x01, 0x32, 0x02, 0x1A, 0xC6, 0x4B, 0xC7, 0x1B, 0x68, 0x33, 0xEE, 0xDF,
 0x03,
 0x64, 0x04, 0xE0, 0x0E, 0x34, 0x8D, 0x81, 0xEF, 0x4C, 0x71, 0x08, 0xC8, 0xF8, 0x69, 0x1C, 0xC1,
 0x7D, 0xC2, 0x1D, 0xB5, 0xF9, 0xB9, 0x27, 0x6A, 0x4D, 0xE4, 0xA6, 0x72, 0x9A, 0xC9, 0x09, 0x78,
 0x65, 0x2F, 0x8A, 0x05, 0x21, 0x0F, 0xE1, 0x24, 0x12, 0xF0, 0x82, 0x45, 0x35, 0x93, 0xDA, 0x8E,
 0x96, 0x8F, 0xDB, 0xBD, 0x36, 0xD0, 0xCE, 0x94, 0x13, 0x5C, 0xD2, 0xF1, 0x40, 0x46, 0x83, 0x38,
 0x66, 0xDD, 0xFD, 0x30, 0xBF, 0x06, 0x8B, 0x62, 0xB3, 0x25, 0xE2, 0x98, 0x22, 0x88, 0x91, 0x10,
 0x7E, 0x6E, 0x48, 0xC3, 0xA3, 0xB6, 0x1E, 0x42, 0x3A, 0x6B, 0x28, 0x54, 0xFA, 0x85, 0x3D, 0xBA,
 0x2B, 0x79, 0x0A, 0x15, 0x9B, 0x9F, 0x5E, 0xCA, 0x4E, 0xD4, 0xAC, 0xE5, 0xF3, 0x73, 0xA7, 0x57,
 0xAF, 0x58, 0xA8, 0x50, 0xF4, 0xEA, 0xD6, 0x74, 0x4F, 0xAE, 0xE9, 0xD5, 0xE7, 0xE6, 0xAD, 0xE8,
 0x2C, 0xD7, 0x75, 0x7A, 0xEB, 0x16, 0x0B, 0xF5, 0x59, 0xCB, 0x5F, 0xB0, 0x9C, 0xA9, 0x51, 0xA0,
 0x7F, 0x0C, 0xF6, 0x6F, 0x17, 0xC4, 0x49, 0xEC, 0xD8, 0x43, 0x1F, 0x2D, 0xA4, 0x76, 0x7B, 0xB7,
 0xCC, 0xBB, 0x3E, 0x5A, 0xFB, 0x60, 0xB1, 0x86, 0x3B, 0x52, 0xA1, 0x6C, 0xAA, 0x55, 0x29, 0x9D,
 0x97, 0xB2, 0x87, 0x90, 0x61, 0xBE, 0xDC, 0xFC, 0xBC, 0x95, 0xCF, 0xCD, 0x37, 0x3F, 0x5B, 0xD1,
 0x53, 0x39, 0x84, 0x0C, 0x14, 0x3C, 0x44, 0xCC, 0x4F, 0xD1, 0x68, 0xB8, 0xD3, 0x6E, 0xB2, 0xCD,
 0x4C, 0xD4, 0x67, 0xA9, 0xE0, 0x3B, 0x4D, 0xD7, 0x62, 0xA6, 0xF1, 0x08, 0x18, 0x28, 0x78, 0x88,
 0x83, 0x9E, 0xB9, 0xD0, 0x6B, 0xBD, 0xDC, 0x7F, 0x81, 0x98, 0xB3, 0xCE, 0x49, 0xDB, 0x76, 0x9A,
 0xB5, 0xC4, 0x57, 0xF9, 0x10, 0x30, 0x50, 0xF0, 0x0B, 0x1D, 0x27, 0x69, 0xBB, 0xD6, 0x61, 0xA3,
 0xFE, 0x19, 0x2B, 0x7D, 0x87, 0x92, 0xAD, 0xEC, 0x2F, 0x71, 0x93, 0xAE, 0xE9, 0x20, 0x60, 0xA0,
 0xFB, 0x16, 0x3A, 0x4E, 0xD2, 0x6D, 0xB7, 0xC2, 0x5D, 0xE7, 0x32, 0x56, 0xFA, 0x15, 0x3F, 0x41,
 0xC3, 0x5E, 0xE2, 0x3D, 0x47, 0xC9, 0x40, 0xC0, 0x5B, 0xED, 0x2C, 0x74, 0x9C, 0xBF, 0xDA, 0x75,
 0x9F, 0xBA, 0xD5, 0x64, 0xAC, 0xEF, 0x2A, 0x7E, 0x82, 0x9D, 0xBC, 0xDF, 0x7A, 0x8E, 0x89, 0x80,
 0x9B, 0xB6, 0xC1, 0x58, 0xE8, 0x23, 0x65, 0xAF, 0xEA, 0x25, 0x6F, 0xB1, 0xC8, 0x43, 0xC5, 0x54,
 0xFC, 0x1F, 0x21, 0x63, 0xA5, 0xF4, 0x07, 0x09, 0x1B, 0x2D, 0x77, 0x99, 0xB0, 0xCB, 0x46, 0xCA,
 0x45, 0xCF, 0x4A, 0xDE, 0x79, 0x8B, 0x86, 0x91, 0xA8, 0xE3, 0x3E, 0x42, 0xC6, 0x51, 0xF3, 0x0E,
 0x12, 0x36, 0x5A, 0xEE, 0x29, 0x7B, 0x8D, 0x8C, 0x8F, 0x8A, 0x85, 0x94, 0xA7, 0xF2, 0x0D,
 0x67, 0x4A, 0xED, 0xDE, 0xC5, 0x31, 0xFE, 0x18, 0x0D, 0x63, 0x8C, 0x80, 0xC0, 0xF7, 0x70, 0x07
 $\};$

$E = \{$
 0x01, 0x03, 0x05, 0x0F, 0x11, 0x33, 0x55, 0xFF, 0x1A, 0x2E, 0x72, 0x96, 0xA1, 0xF8, 0x13,
 0x35,
 0x5F, 0xE1, 0x38, 0x48, 0xD8, 0x73, 0x95, 0xA4, 0xF7, 0x02, 0x06, 0x0A, 0x1E, 0x22, 0x66, 0xAA,
 0xE5, 0x34, 0x5C, 0xE4, 0x37, 0x59, 0xEB, 0x26, 0x6A, 0xBE, 0xD9, 0x70, 0x90, 0xAB, 0xE6, 0x31,
 0x53, 0xF5, 0x04, 0x0C, 0x14, 0x3C, 0x44, 0xCC, 0x4F, 0xD1, 0x68, 0xB8, 0xD3, 0x6E, 0xB2, 0xCD,
 0x4C, 0xD4, 0x67, 0xA9, 0xE0, 0x3B, 0x4D, 0xD7, 0x62, 0xA6, 0xF1, 0x08, 0x18, 0x28, 0x78, 0x88,
 0x83, 0x9E, 0xB9, 0xD0, 0x6B, 0xBD, 0xDC, 0x7F, 0x81, 0x98, 0xB3, 0xCE, 0x49, 0xDB, 0x76, 0x9A,
 0xB5, 0xC4, 0x57, 0xF9, 0x10, 0x30, 0x50, 0xF0, 0x0B, 0x1D, 0x27, 0x69, 0xBB, 0xD6, 0x61, 0xA3,
 0xFE, 0x19, 0x2B, 0x7D, 0x87, 0x92, 0xAD, 0xEC, 0x2F, 0x71, 0x93, 0xAE, 0xE9, 0x20, 0x60, 0xA0,
 0xFB, 0x16, 0x3A, 0x4E, 0xD2, 0x6D, 0xB7, 0xC2, 0x5D, 0xE7, 0x32, 0x56, 0xFA, 0x15, 0x3F, 0x41,
 0xC3, 0x5E, 0xE2, 0x3D, 0x47, 0xC9, 0x40, 0xC0, 0x5B, 0xED, 0x2C, 0x74, 0x9C, 0xBF, 0xDA, 0x75,
 0x9F, 0xBA, 0xD5, 0x64, 0xAC, 0xEF, 0x2A, 0x7E, 0x82, 0x9D, 0xBC, 0xDF, 0x7A, 0x8E, 0x89, 0x80,
 0x9B, 0xB6, 0xC1, 0x58, 0xE8, 0x23, 0x65, 0xAF, 0xEA, 0x25, 0x6F, 0xB1, 0xC8, 0x43, 0xC5, 0x54,
 0xFC, 0x1F, 0x21, 0x63, 0xA5, 0xF4, 0x07, 0x09, 0x1B, 0x2D, 0x77, 0x99, 0xB0, 0xCB, 0x46, 0xCA,
 0x45, 0xCF, 0x4A, 0xDE, 0x79, 0x8B, 0x86, 0x91, 0xA8, 0xE3, 0x3E, 0x42, 0xC6, 0x51, 0xF3, 0x0E,
 0x12, 0x36, 0x5A, 0xEE, 0x29, 0x7B, 0x8D, 0x8C, 0x8F, 0x8A, 0x85, 0x94, 0xA7, 0xF2, 0x0D,
 0x67, 0x4A, 0xED, 0xDE, 0xC5, 0x31, 0xFE, 0x18, 0x0D, 0x63, 0x8C, 0x80, 0xC0, 0xF7, 0x70, 0x07
 $\};$

Addition cyclic key

In this operation, a round key is added to the state using a simple EXOR. The round key is derived from the encryption key using a key derivation algorithm (key schedule).

Length cyclic key equal length block Nb.



Rice. 4. 5. At addition key cyclic key folds via EXOR with state

When encrypting, parts of the extended key are selected from the beginning to the end, when deciphering - from the end to the beginning.

Extension key (Key Expansion)

The extended key is a linear array of 4-byte words and is denoted as $W[Nb*(Nr+1)]$. The first Nk words contain the encryption key. All remaining words are determined recursively from words with lower indices. The key generation algorithm depends on the value of Nk : below is a version for Nk equal to or less than 6 and a version for Nk greater than 6.

For $Nk < 6$ or $Nk = 6$ we have:

```
KeyExpansion(CipherKey,W)
{
    for (i = 0; i < Nk; i++) W[i] = CipherKey[i];
    for (j = Nk; j < Nb*(Nk+1); j += Nk)
    {
        W[j] = W[j-Nk] ^ SubByte( Rotl( W[j-1] ) ) ^ Rcon[j/Nk];

        for (i = 1; i < Nk && i+j < Nb*(Nr+1); i++)
            W[i+j] = W[i+j-Nk] ^ W[i+j-1];
    }
}
```

As can be seen, the first Nk words are filled with the encryption key. Each subsequent word $W[i]$ is obtained by EXORing the previous word $W[i-1]$ and the word Nk positions earlier, $W[i-Nk]$. For words whose position is a multiple of Nk , a transformation to $W[i-1]$ is applied before the EXOR, and then a cycle constant is added. The transformation consists of a cyclic shift of bytes in the word, denoted as $Rotl$, followed by $SubByte$, which is the application of byte substitution.

For $Nk > 6$ We we have :

```
KeyExpansion(CipherKey,W)
{
    for (i=0; i<Nk; i++) W[i]=CipherKey[i];
    for (j=Nk; j<Nb*(Nk+1); j+=Nk)
    {
        W[j] = W[j-Nk] ^ SubByte(Rotl(W[j-1])) ^ Rcon[j/Nk];
        for (i=1; i<4; i++) W[i+j] = W[i+j-Nk] ^ W[i+j-1];
        W[j+4] = W[j+4-Nk] ^ SubByte(W[j+3]);
    }
}
```

```

for (i=5; i<Nk; i++) W[i+j] = W[i+j-Nk] ^ W[i+j-1];
}
}

```

The difference For schemes at $Nk > 6$ consists of V application SubByte for every 4th byte of Nk.

The cycle constant is independent of Nk and is defined as follows :

```

Rcon[i] = ( RC[i], '00' , '00' , '00' ), Where
RC[0]='01'
RC[i]=xtime(Rcon[i- 1])

```

Encryption

Cipher Rijndael includes next transformations:

- initial addition cyclic key;
- Nr-1 cycles;
- final cycle.

On pseudo-C This looks like next way:

```

Rijndael (State, CipherKey)
{
KeyExpansion(CipherKey, ExpandedKey); // Key expansion
AddRoundKey(State, ExpandedKey); // Addition cyclic key For ( i=1 ;
i<Nr ; i++) Round(State,ExpandedKey+Nb*i); // cycles
FinalRound(State, ExpandedKey+Nb*Nr); // final cycle
}

```

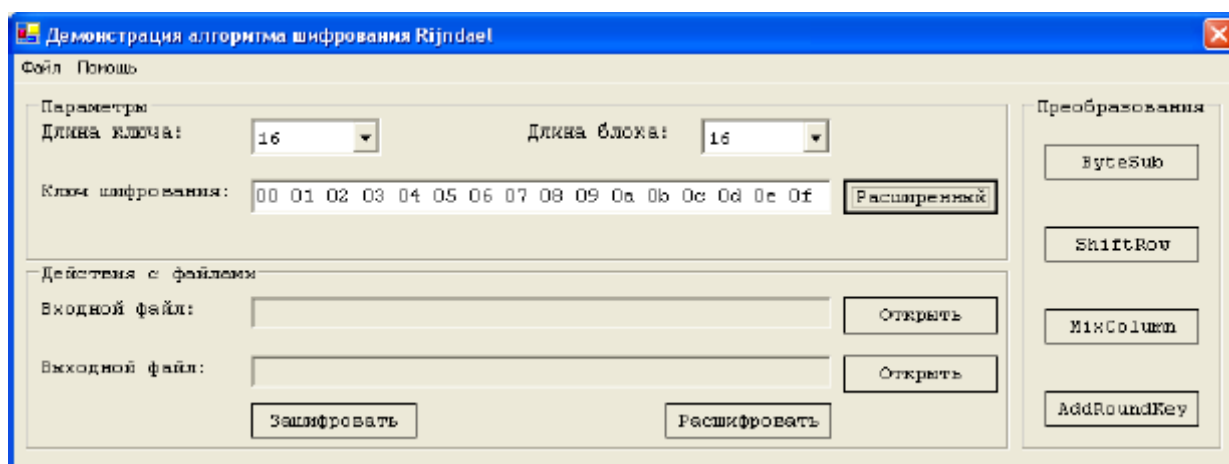
If preliminary completed procedure extensions key, then the procedure will look like this:

```

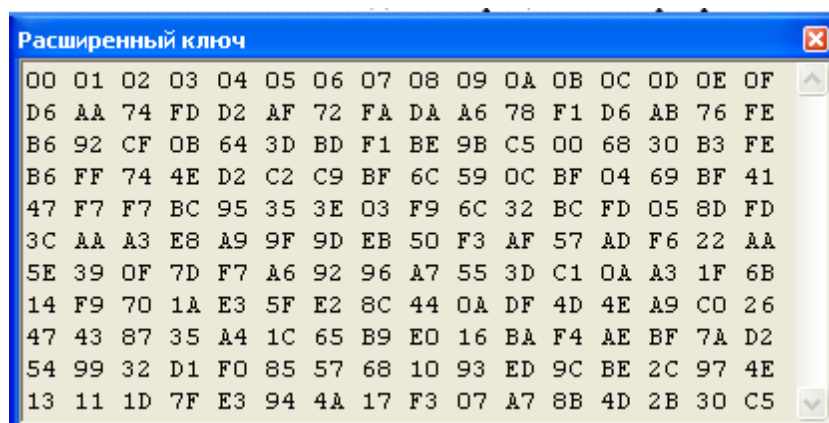
Rijndael (State, CipherKey)
{
AddRoundKey(State, ExpandedKey);
For ( i=1 ; i<Nr ; i++) Round(State,ExpandedKey+Nb*i); FinalRound(State, ExpandedKey+Nb*Nr);
}

```

Description of the demo program The program is completed The program is written in C# and consists of two elements: the Rijndael.dll file, which contains the implementation of the encryption algorithm, and the RijndaelDemo.exe demo application. The application requires Windows OS with the .NET Framework v1.1 installed. In the demo program's main window, the key length and block length are set, and the extended encryption key, calculated in accordance with the specified encryption key, can be viewed.

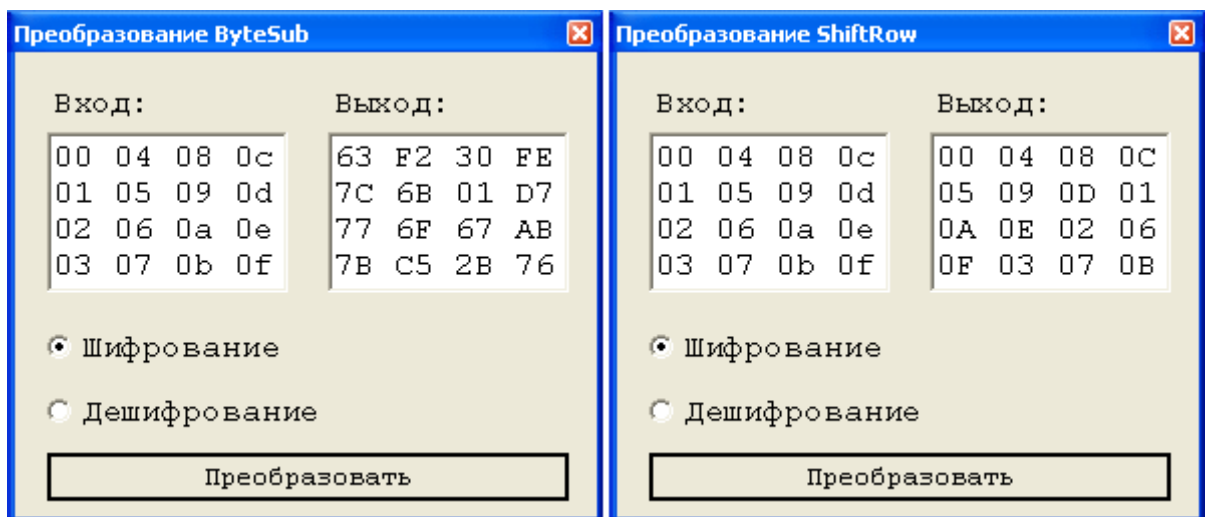


Drawing 2. Main window demonstration programs

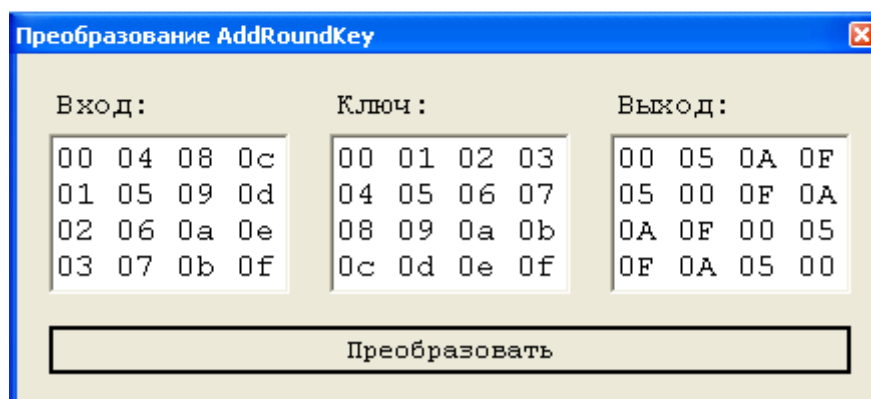
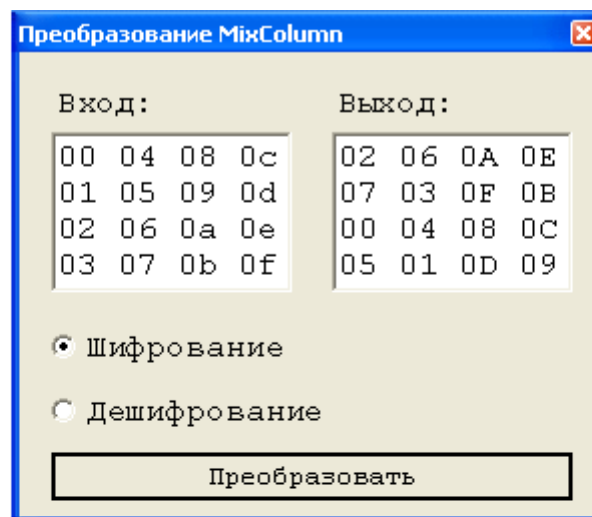


Drawing 3. Window extended key

It is also possible to examine in detail the action of all cyclic transformations (ByteSub, ShiftRow, MixColumn, AddRoundKey) both during encryption and decryption, Fig. 4, 5.



Drawing 4. Windows transformations ByteSub And ShiftRow



Drawing 5. Windows transformations MixColumn And AddRoundKey

When encrypting, you are asked to select the source file and the file where the encryption result will be placed, and when decrypting, respectively. encrypted file And file, designed For premises result-

Decryption data. The encryption key, key length, and block length specified in the main program window are used during the process.

4. Equipment And materials

For execution work you will need:

- Class personal computer Not below Pentium-4, cool board, chalk;
- operating room system Microsoft Windows, software tool

RijndaelDemo ,

5. Instructions By technology security

Requirements security before beginning works:

- PROHIBITED: changing clothes, using fire, smoking, eating in the laboratory;
- Ensure that electrical outlets and connectors are intact. Do not operate, break, or attempt to repair faulty power tools.
- V laboratories necessary be V replaceable shoes;
- Turn on the computer only after receiving permission for the work being performed and the permission of the teacher or laboratory assistant.

Requirements security in time works:

- When performing laboratory work, students are required to use only computers, peripheral equipment, connecting cables, measuring equipment and diskettes that are directly related to the given laboratory lesson;
- connection and disconnection of components of the computing complex should be carried out only with complete removal of tension from all components computing complex.

Requirements security By at the end works:

- at the teacher's command, students are required to turn off the computer and components of the computing complex;

- working place bring V order.

6. Tasks

Task No. 1. Read the information about the *RijndaelDemo program* , described in section 3. Run the *RijndaelDemo.exe module* .

Task No. 2. Using regular text files as an example, study encryption and decryption methods using the *Rijndael algorithm* . Examine in detail the operation of all cyclic transformations (ByteSub, ShiftRow, MixColumn, AddRoundKey), both during encryption and decryption. The source text for encryption can be prepared in advance and saved in a **.txt file* .

Task No. 3. Save screen forms in the report demonstrating the process of encryption and decryption of information, analyze the obtained results.

Task No. 4. Include in the laboratory report the answers to the control questions selected in accordance with the option number specified by the instructor.

7. Content report

Report must contain:

1. Name work, target.
2. Subsequence actions at execution each tasks.
3. Conclusions By execution laboratory work.

8. Control questions

Номер варианта	Контрольные вопросы
1,5,7,26	Сравните основные характеристики алгоритмов <i>Rijndael</i> и ГОСТ 28147-89.
2,4,6	Сравните основные характеристики алгоритмов <i>Rijndael</i> и <i>DES</i> .
11,13	Опишите структуру сети Фейстеля.
12,14,16	Приведите обобщенные схемы шифрования данных с помощью алгоритма <i>Rijndael</i> и ГОСТ 28147-89. Дайте их сравнительный анализ.
3,9,18,29	Сравните один раунд шифрования данных с помощью алгоритма <i>Rijndael</i> и ГОСТ 28147-89.
20,22,24	Сравните эквивалентность прямого и обратного преобразований в алгоритмах <i>Rijndael</i> и ГОСТ 28147-89.
10,17,19	Сравните выработку ключевой информации в алгоритмах <i>Rijndael</i> и ГОСТ 28147-89.
21,23,25	Сравните алгоритмы <i>Rijndael</i> и ГОСТ 28147-89 по показателям диффузии.
8, 28,27	Сравните алгоритмы <i>Rijndael</i> и ГОСТ 28147-89 по показателям стойкости.
12,15,30	Сравните алгоритмы <i>Rijndael</i> и ГОСТ 28147-89 по показателям производительности и удобству реализации.

9. A list of recommended literature on this topic

1. Zhuk, A.P., Zhuk, E.P., Lepeshkin, O.M., Timoshkin, A.I. Information Security: A Textbook for Bachelors [Text] / A.P. Zhuk et al. - M.: INFRA-M, 2015. -392 p.

2. Shan'gin, V.F. Computer Information Protection. Effective Methods and Tools [Text] / V.F. Shan'gin – M.: DMK Press, 2012. – 544 p.

LABORATORY JOB No. 4

STUDYING DEVICES AND PRINCIPLE WORK OF THE *ENIGMA* CIPHER MACHINE

Objective: To study the principles of encryption/decryption used in the Enigma encryption machine. To become familiar with the general operating principles of the Enigma encryption machine using an emulator. The *Enigma3S emulator must first be installed* .

1. Description of the laboratory work

Enigma, device And principle actions



Rice. 3.1. External view encryption cars Enigma

Enigma (*Enigma*) ⁴ is a portable cipher machine used to encrypt and decrypt secret messages. More precisely, Enigma — whole family electromechanical rotary cars, applied With 20s years XX centuries.

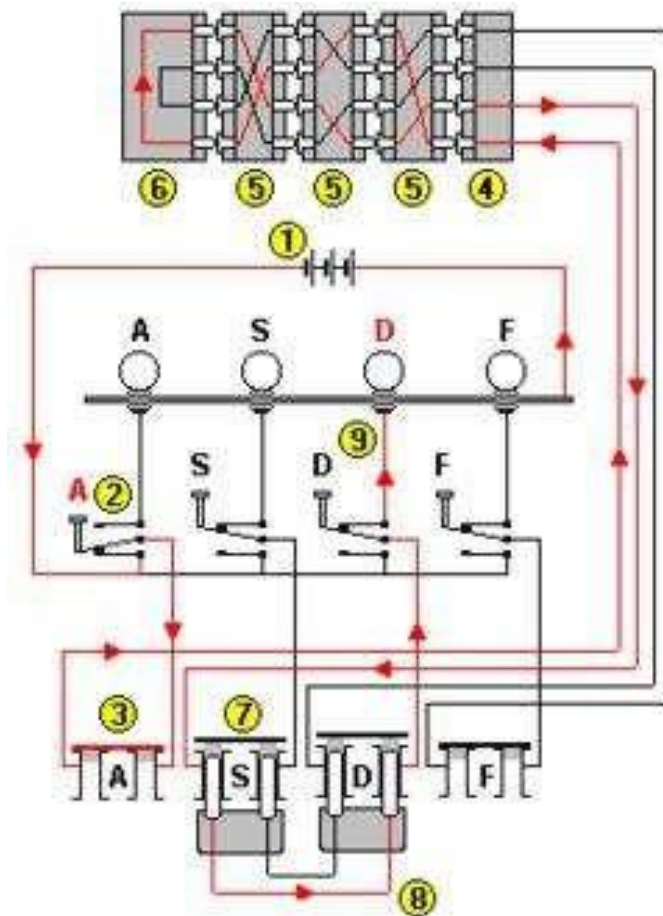
⁴ Emulators Enigma <http://www.attlabs.att.co.uk/andyc/enigma>

The Enigma was used commercially, as well as in military and government services, in many countries around the world, but it was most widely used in Germany during World War II. It is the *Wehrmacht Enigma* —the German military model—that is most often studied.

Although the Enigma cipher was, from a cryptographic point of view, quite weak, in practice only a combination of this factor with others, such as operational errors,

rators, procedural flaws And captures copies Enigma

And code books, allowed English cryptanalysts to crack messages encrypted with the Enigma cipher.



Rice. 3.2. Electric scheme Enigma (replacement V text letters 'A' letter 'D')

The encryption action of the Enigma is shown for two keys pressed in sequence - the current flows through the rotors, is "reflected" from the reflector, and then again returns through rotors. Gray lines on drawing Other possible electrical circuits within each rotor are shown. The letter "A" is replaced in the ciphertext differently with successive keystrokes, first by "G", then by "C". The signal takes a different route due to the rotation of the rotor.

Like other rotor machines, the Enigma consisted of a combination of mechanical and electrical systems. The mechanical part included a keyboard, a set of rotating disks (rotors) located along a shaft, and a stepping mechanism that set one or more rotors in motion with each keystroke. The movement of the rotors results in various character substitutions at each next keystroke on the keyboard.

The mechanical parts moved, forming a changing electrical circuit—that is, the encryption was essentially carried out electrically. When keys were pressed, the circuit was closed, current flowed through the various components, and ultimately lit one of many lights, displaying the letter being output. For example, when encrypting a message beginning "ANX...", the operator would first press the "A" key, and the "Z" light would light, indicating that "Z" became the first letter of the cryptogram. The operator would continue encryption by pressing "N" on the keyboard, and so on, until the end of the original message.

The constant change in the electrical circuit through which the current flowed, due to the rotation of the rotors, made it possible to implement a polyalphabetic substitution cipher, which provided high security for that time.

Rotors



Rice. 3.3. Left side rotor Enigmas, visible flat electrical contacts



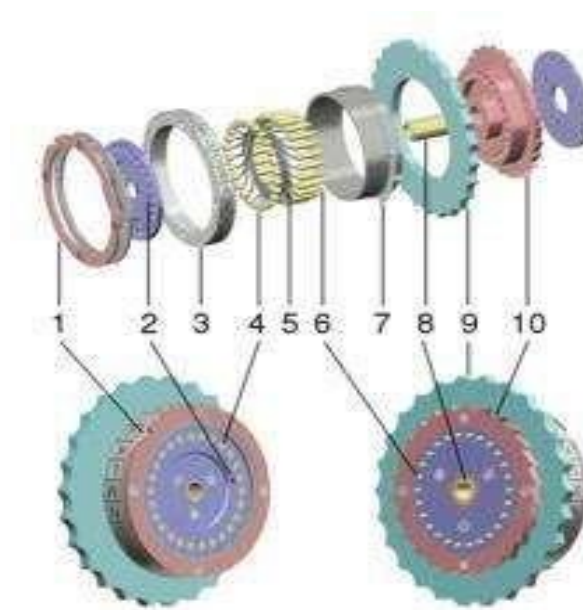
Rice. 3.4. Right side rotor, visible pin contacts

The rotors are the heart of the Enigma machine. Each rotor is a disk approximately 10 cm in diameter, made of hard rubber or Bakelite, with spring-loaded pin contacts on one side of the rotor, arranged in a circle; on the other side is a corresponding number of flat electrical contacts. The pins and flat contacts correspond to letters of the alphabet; typically, these are the 26 letters “A” through “Z.” When the contacts of adjacent rotors touch, they complete an electrical circuit. Inside the rotor, each pin contact is connected to a flat contact. The order of the connections can vary.

The rotor itself reproduces the encryption by simply replacing the symbols. For example, the contact responsible for the letter "E" can be connected to the contact letters "T" on another side rotor. But at use Not-

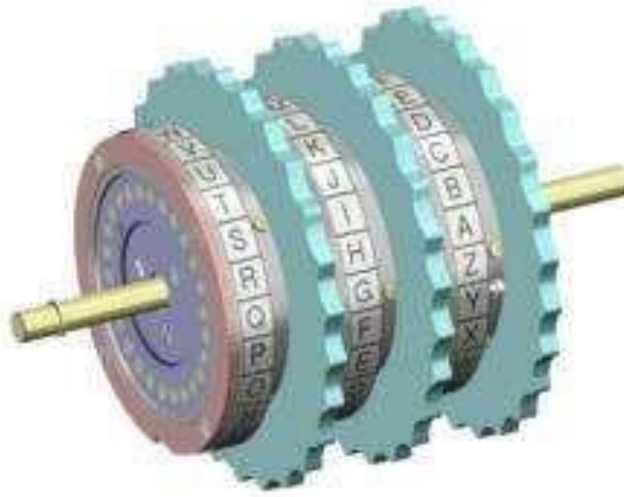
The number of rotors in a chain (usually three or four) results in a more robust type of polyalphabetic encryption due to their constant movement.

The rotor can occupy one of 26 positions in the machine. It can be turned manually by means of a knurled finger wheel that protrudes outward, as shown in Fig. 3.5. So that the operator can always determine position rotor, on each rim located alphabetical ring; One of the letters is visible through the window. In early Enigma models, the alphabet ring was fixed; later versions introduced a more complex design with the ability to adjust it. Each rotor contains a notch (or several notches) used to control the movement of the rotors.



Rice. 3.5. Rotor V disassembled view

1. ring with notches
2. marking dot For contact "A"
3. alphabetical ring
4. tinned contacts
5. wiring
6. pin contacts
7. spring lever For settings rings
8. sleeve
9. finger ring
10. ratchet wheel



Rice. 3.6. Three sequentially united rotor

Military versions of the Enigma were produced with multiple rotors; the first model contained only three. In 1938, this was increased to five, but only three were used in the machine at any one time. These rotor types were marked with the Roman numerals I, II, III, IV, and V, all with a single notch located in different places on the alphabet ring. Naval versions of *the Wehrmacht Enigma* contained more rotors than other models: six, seven, or eight.

In *the Wehrmacht Enigma*, each rotor was attached to an adjustable ring with notches. The five basic rotors (IV) each had one notch,

Then How naval With additional rotors (VI-VIII) — two at a time. At a certain point, the notch falls opposite the pawl, allowing it to engage the ratchet of the next rotor when the key is pressed again. When same dog Not hits V recess, she just slips through along the surface of the ring without engaging the gears. In a single-notch system, the second rotor advances one position while the first advances 26. Similarly, the third rotor advances one step while the second advances 26 steps. A special feature was that the second rotor also turned if the third one turned; this meant that the second rotor could turn twice with two consecutive keystrokes, a so-called "two-step motion," which resulted in a shorter encryption period.

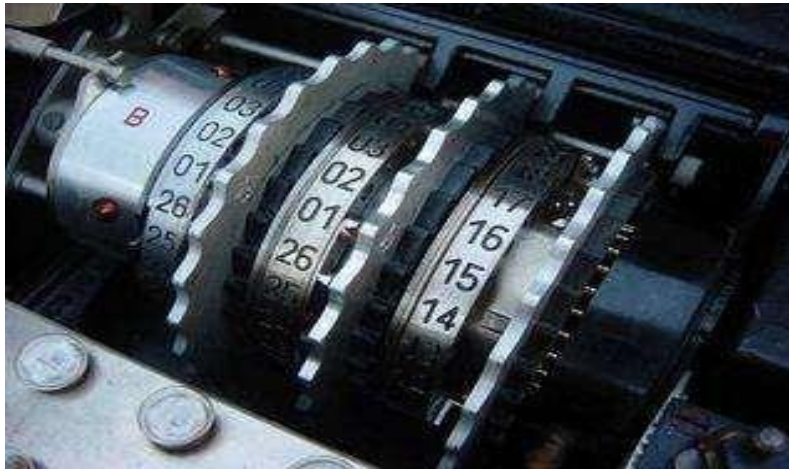


Fig. 3.7. Enigma rotors in assembled condition Three movable rotor placed between two motionless details: input ring And reflector (marked "B" on the left)

Input wheel

The input wheel (*Eintrittswalze* in German), or input stator, connects the plugboard, or (if there is none) the keyboard and lamp board, to the rotors. Although the fixed connection of wires is of relatively little importance from a security standpoint, it proved somewhat of an obstacle in the work of the Polish cryptanalyst Marian Redzewski when he was trying to determine the way the wires were connected inside the rotors. The commercial version of Enigma connected the letters in the order they appeared on the keyboard: QA, WB, EC, and so on. However, The military version connected them in direct alphabetical order: AA, BB, CC, etc.

Reflector

With the exception of the early A and B models, the last rotor was followed by a reflector (*Umkehrwalze* in German), a patented feature that distinguished the Enigma family from other rotor machines developed at the time. The reflector connected the contacts of the last rotor in pairs, commutating the current through the rotors in the opposite direction, but in a different direction. route. Reflector guarantees, What transformation, feasible Enigma, There is involution-

tion, i.e., the decryption process is symmetrical to the encryption process. In addition, the reflector gives Enigma the property that no letter can be encrypted with itself. This

was a serious conceptual flaw that was later exploited by codebreakers.

Switching panel



Fig. 3.8. The patch panel at the front of the machine. Up to 13 connections could be used. In the photo, two pairs of letters are switched (“S”-“O” and “J”-“A”)

The plugboard, which allowed the operator to vary the wiring connections, first appeared in German military versions in 1930 and was soon used successfully in naval versions. The plugboard made a huge contribution to the complexity of the Enigma cipher, even more so than the addition of the additional rotor. An Enigma without a plugboard— an *"unsteckered"* Enigma—could be deciphered almost entirely by hand, but these methods proved useless when a plugboard was added, and crackers were forced to construct special deciphering machines. A cable placed on the plugboard connects letters in pairs, so that "E" and "Q" might be paired. The effect is to permute these letters before and after the signal passes through the rotors. For example, when the operator pressed "E," the signal would be routed to "Q," and only then to the input rotor. At the same time, it could several such pairs can be used (up to 13).

Accessories

A handy feature used on the M4 Enigma was the "*Schreibmax*", a small printing device that could print all 26 letters on a small sheet of paper.

Procedures use Enigma

During World War II, German operators used a codebook only to set the rotors and ring settings. For each message, they selected a random starting position, say "WZA," and a random message key, say "SXT." The operator then set the rotors to the starting position "WZA" and encrypted the message key "SXT." Let's assume that the encrypted key would result in "UHL." The operator then set the message key "SXT" as the starting position of the rotors and encrypted the message. The operator then sent the starting position "WZA" and the encrypted key "UHL" along with the message. The recipient set the starting position according to the first trigram "WZA" and decrypted the second trigram "UHL." Provide the answers to the control questions in the report according to the option number specified by the teacher.

The original "SXT" key was recognized. The recipient then used this key as a starting point to decrypt the message. Keys typically expired within one day.

Military version Enigma used only 26 letters. Others symbols were replaced by rare letter combinations. The space was omitted or replaced by "X". The "X" symbol was also used to indicate a period or the end of a message. Some special symbols were used

V In some armed units, for example, *the Wehrmacht* replaced the comma with two symbols "ZZ" and a question mark - "FRAGE" or "FRAQ".

The *Kriegsmarine* replaced the comma with “Y” and the question mark with “UD”. Two, three or four zeros were replaced by CENTA, MILLE and MYRIA respectively .

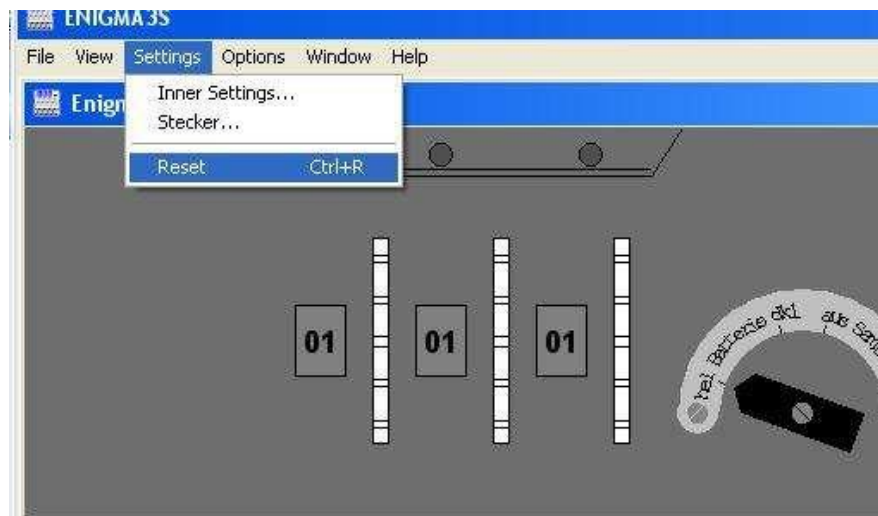
When performing laboratory work to study the Enigma cipher, the Enigma3S emulator program is used .

2. Exercise

1. Launch the *Enigma3S emulator* from the folder specified by the teacher.

Check the help file: menu option

Help=>Help. IN menu programs select paragraph Settings => Reset, Fig.3.9.

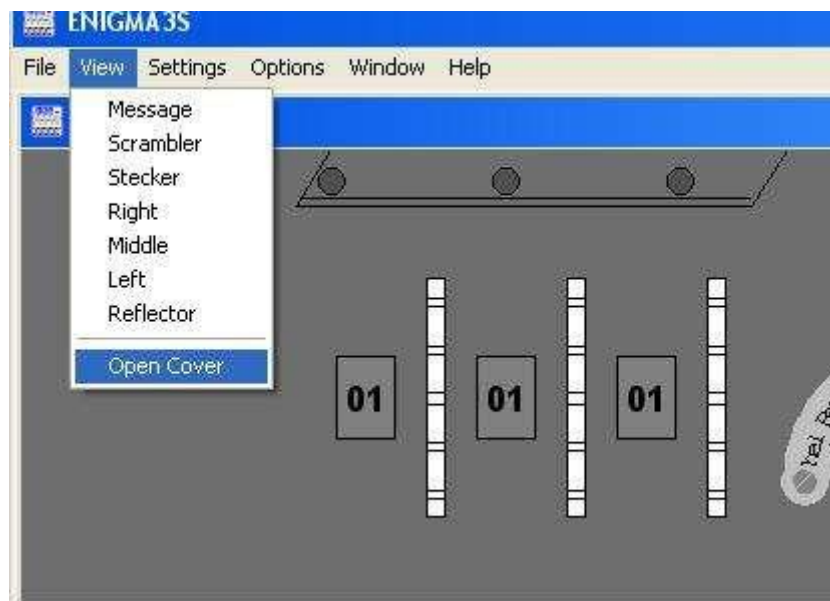


Rice. 3.9. Initial installations emulator Enigma

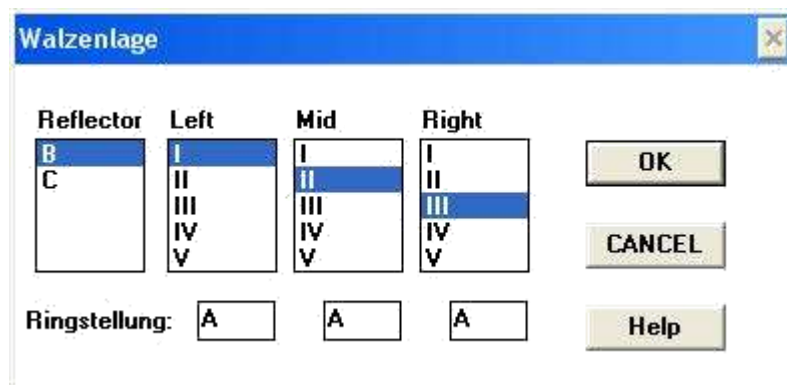
3. Install values For rings 01 01 01 by way choice menu item

View=>Open Cover. In the menu Settings=>Inner Settings set the following values: Reflector – B, Left – I, Middle – II, Right – III, Ringstellung – AAA (We will consider this position as the initial one),

Fig. 3.10, 3.11.



Rice. 3.10. View initial installations



Rice. 3.11. Installations Reflector, Left, Mid, Right

4. Enter "SECRET MESSAGE" on the Enigma keyboard. What message is returned?
5. Repeat paragraph 3, having changed settings Ringstellung – AAA on Ringstellung – ABC
6. Enter the message “SECRET MESSAGE” on the Enigma keyboard. What message is obtained at the output? How does it differ from the message obtained in step 4?
7. Save the ciphertext obtained in step 6 using the menu option File=>Save CT as, Fig. 3.12.



Rice. 3.12. Preservation ciphertext

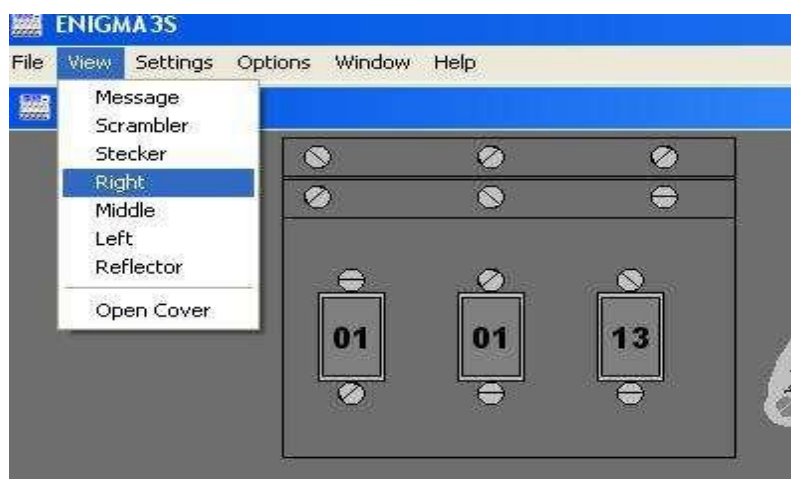
8. Create a file with the .pln extension in the root of the folder with the emulator program, open it for editing in Notepad and write the plaintext for encryption into it.

9. In the program menu, select the File=>Open Pt option and select the file created in step 9. Obtain the ciphertext from the plaintext by selecting the Options=>Encipher Text menu option.

10. Set the emulator to its initial position. In the program menu, select Window => Scrambler. Using the keyboard, enter a random 22-character message (options are listed in *Table 3.1*), paying attention to the position of the rings. Enter the 22-character sequence again. How has the position of the rings changed? Is the new encrypted sequence different from the initial one? Why has the output sequence changed?

11. Using the Scrambler window, observe how the ciphertext changes depending on the contact wheel position settings. What does it mean to customize the contact wheel order?

IN window programs select option View=>Right, rice. 3.13.



Rice. 3.13. View installations For execution tasks

Fill in table conformity For 5 letters options indicated in table 3.1) for the first 12 angular positions of the right wheel.

Table 3.1

Letters on at the entranc e	Corner wheel position											
	11	22	33	44	55	66	77	88	99	110	111	112

13. Knowing that the emulator is set to the initial position, decipher text in accordance with the option number specified by the teacher (options for test tasks are indicated in *tables 3.2, 3.3, 3.4*).

Table 3.2

Numbers options	Original alphabet
1,5,9,13,17	QRTYU FBNAK GHERL ADLKE DS
2,6,10,14,18,22	UIERT PAEVC DSNCY OPLKD BV
3,7,11,15,19,23,27	UIFGH KLBVQ FDIIT QKJLS DB
4,8,12,16,20,24,28	LLWER TYYWV BAFDP WRTOPF JK
21,25,29,26,30	OPJHG JFDPJ GFDSK LDFHU BX

Table 3.3

Numbers options	Ciphertext
1,5,9,13,17	ABCDE
2,6,10,14,18,22	FGHIJK
3,7,11,15,19,23,27	LMNOP
4,8,12,16,20,24,28	QRSTU
21,25,29,26,30	VWXYZ

Table 3.4

Numbers options	Ciphertext
1,5,9,13,17	FQGAH WABUN NL
2,6,10,14,18,22	QIKOL RCRJS EGBSS X
3,7,11,15,19,23,27	OOKWE PRFMI M
4,8,12,16,20,24,28	KIXDI ACTHJ L
21,25,29,26,30	XLXOO EABUN NL

Laboratory Job No. 5

Generation simple numbers, used V asymmetrical system- encryption max

Target works : research generation simple numbers, used in public key encryption systems and the ability to draw independent conclusions based on research results.

2. Formed competencies

This laboratory work is aimed at developing the following competencies of a bachelor's degree graduate:

OPK-1: the ability to install software and hardware for information and automated systems;

PC-5: the ability to interface hardware and software within information and automated systems.

3. Theoretical Part

Asymmetrical systems encryption

Asymmetric cryptosystems (systems open encryption, public key systems) - the meaning of these cryptosystems is that different transformations are used for encryption and decryption. One of them - encryption - is absolutely open to everyone. The other same – decryption remains secret due to the secrecy of the decryption key. Thus, anyone who wants to encrypt something uses an open transformation. But only someone who has the secret key can decrypt and read it. The diagram of an asymmetric cryptosystem is shown in Fig. 1.1.

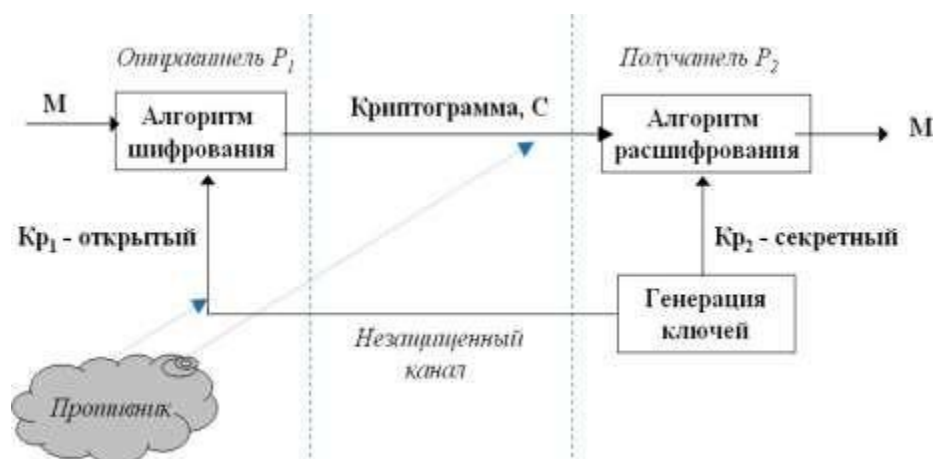


Рис. 1.1. Generalized scheme asymmetrical cryptosystems

Currently, in many asymmetric cryptosystems, the type of transformation is determined by the key. The user has two keys – a secret key and a public key. The public key is published in a public place, and anyone who wants to send a message to this user encrypts the text with the public key. Only the mentioned user with the secret key can decrypt it. This eliminates the problem of transmitting the secret key, as in symmetric systems. However, despite all their advantages, these cryptosystems are quite labor-intensive and slow. The security of asymmetric cryptosystems is based primarily on the algorithmic difficulty of solving a problem in an acceptable time. If an attacker succeeds in constructing such an algorithm, the entire system and all messages encrypted using this system will be discredited. This is the main danger of asymmetric cryptosystems, as opposed to symmetric ones.

Algorithm Diffie- Hellman

Diffie -Hellman algorithm uses a discrete exponentiation function. First, two large prime numbers are generated. n And q . These two numbers Not Necessarily keep V secret. Next one from

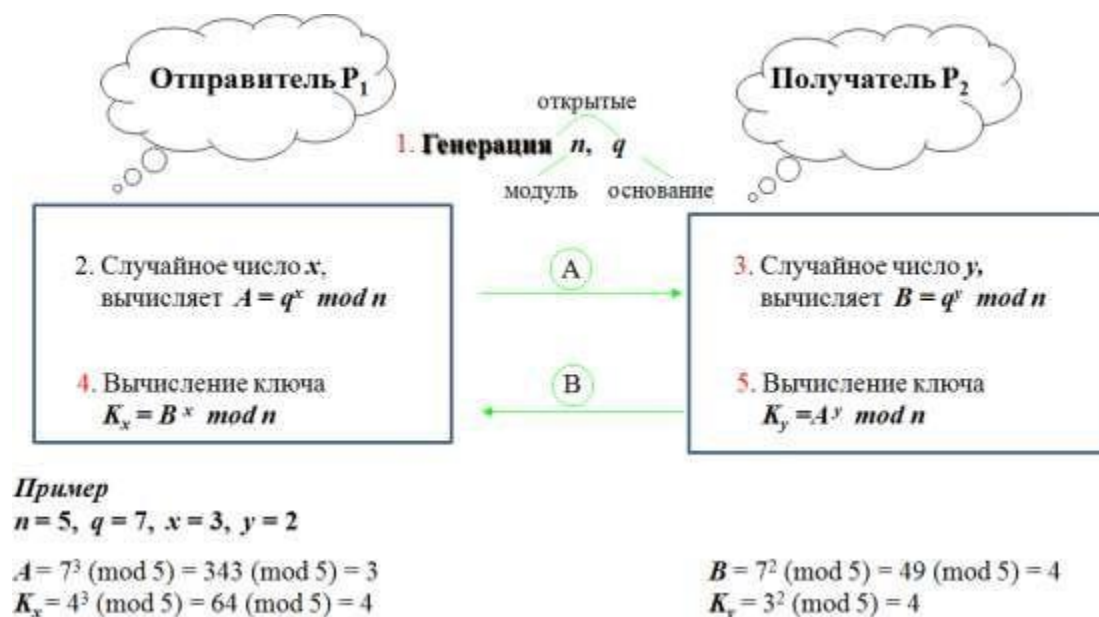
partners P_1 generates a random number x and sends it to another participant in future exchanges P_2 meaning

$$A = q^x \bmod n$$

Upon receipt of A partner P_2 generates a random number y and sends it to the exchange participant P_1 calculated value

$$B = q^y \bmod n$$

Partner P_1 , having received B , calculates $K_x = B^x \bmod n$, and partner P_2 calculates $K_y = A^y \bmod n$. The algorithm guarantees that the numbers K_y and K_x equal and can be used as a secret key for encryption. After all, even if we intercept the numbers A and B , it is difficult to calculate K_x or K_y . Schematically, the operation of the Diffie-Hellman algorithm is shown in Fig. 1.2.



Rice. 1.2. Algorithm Diffie- Hellman

Algorithm RSA

The first practical implementation of the principle of open encryption was in the *RSA system*, developed in 1977 at the Massachusetts Institute of Technology (USA) and named after the first letters of the authors' last names: Ronald Rivest (*R.Rivest*), Adi Shamir (*A.Shamir*), Leonard Adleman (*L.Adleman*).

The idea of the authors was that by taking an integer N in the form of a product of two large prime numbers $N=P*Q$, it is easy to select a pair of numbers Y and X such that for any integer M less than N , the following relation is valid :

$$(M^X)^Y = M \bmod N$$

The public encryption key in the *RSA system* is the key Y and the modulus N , and the secret key for decrypting messages is the number X .

The procedure for encrypting a message M considered as an integer (this assumption is possible due to the fact that any content can be presented in numerical form when processed in computer technology), less than N (if necessary, a long message is divided into segments that are encrypted independently), consists of calculating the value :

$$C=M^Y \bmod N$$

Decryption is carried out similarly using the secret key X :

$$M=C^X \bmod N$$

It can be mathematically proven that determining the secret key X from a pair of numbers (N, Y) is no simpler than factoring N into prime factors , that is, finding P and Q . Factoring an integer has been studied in mathematics since ancient times and is known as a difficult computational problem. Currently, factoring a number with several hundred decimal places would require hundreds of years of continuous operation for modern computers.

Schematically, Job algorithm *RSA* presented on rice. 1.3.

Генерация ключей

Получатель 1. P, Q - простые, $N = P \cdot Q$
2. $\varphi(N) = (P-1) \cdot (Q-1)$, $\varphi(N)$ - функция Эйлера

Выбор открытого ключа Y :

$$1 < Y \leq \varphi(N), \text{НОД}(Y, \varphi(N)) = 1$$

Вычисление секретного ключа X :

$$X \cdot Y \equiv 1 \pmod{\varphi(N)}$$

Отправитель **шифрование** M ($M_1 = 0, 1, 2, \dots, N-1$)
3. $C_1 = M_1^Y \pmod{N}$

$(N, Y) \rightarrow$ отправителю

Получатель **расшифрование** C ($C_1, C_2, \dots, C_p, \dots$)
4. $M_1 = C_1^X \pmod{N}$

Пример

Генерация ключей

1. $P = 3, Q = 11, N = P \cdot Q = 33$
2. $\varphi(N) = (P-1) \cdot (Q-1) = 2 \cdot 10 = 20$
 $Y = 7, \text{НОД}(Y, \varphi(N)) = 1$
 $X \cdot Y \equiv 1 \pmod{20}, 7 \cdot 3 \equiv 1 \pmod{20}, X = 3$

Сообщение: $M_1 M_2 \rightarrow 32; M_1 = 3 < 33, M_2 = 2 < 33$

Шифрование $C_1 = M_1^Y \pmod{N}$

3. $C_1 = 3^7 \pmod{33} = 2187 \pmod{33} = 9$
 $C_2 = 2^7 \pmod{33} = 128 \pmod{33} = 29$

Шифротекст 9,29

Расшифрование $M_1 = C_1^X \pmod{N}$

4. $M_1 = 9^3 \pmod{33} = 729 \pmod{33} = 3$
 $M_2 = 29^3 \pmod{33} = 24389 \pmod{33} = 2$

Восстановленный текст 3,2

Rice. 1.3. Algorithm RSA

1.1. Methods checks numbers on simplicity

One of the main problems of asymmetric encryption is the generation of large prime numbers. The simplest method for checking the primality of a natural number N is the trial division method: for $d = 2, 3, 4, \dots$, we check whether the condition $(d, N) > 1$ is satisfied (here (d, N) is the greatest common divisor of d and N). The number of operations required for this method is of the order of the square root of N .

That's why already For numbers order $10^{30} - 10^{40}$ He not applicable.

In contrast to such “deterministic” tests, there are also “probabilistic” ones. tests checks simplicity. For the subject numbers checked

certain conditions related to random numbers must be met. If any of these conditions is not met, then N is a composite

number. If all conditions are met, then we can say with some probability that N is a prime number. This probability is closer to 1 the more random numbers we test.

Typically, these conditions are based on Fermat's little theorem, which states that for any positive number b not exceeding some prime number p :

$$b^{(p-1)} = 1 \pmod{p}.$$

For example, $2^6 = 64 = 63 + 1 = 1 \pmod{7}$. If you want to determine whether an integer r is prime, you can choose any positive integer b less than r and check whether the equality holds.

$$b^{(r-1)} = 1 \pmod{r}.$$

If the equality is not satisfied, then, based on Fermat's theorem, we can be absolutely sure that r is not a prime number. If the equality is satisfied, then we can only assume that r is a prime number and therefore call it “pseudoprime to base b .” The probability $P(x)$ that a composite number x turns out to be pseudoprime to a random base decreases with increasing x .

Unfortunately, there are so-called Carmichael numbers – composite numbers that have the property:

$$b^{(r-1)} = 1 \pmod{r} \text{ for all } b \text{ in the interval } [1, r] \text{ that are coprime to } r .$$

Example numbers Carmichael is number $561 = 3 \cdot 11 \cdot 17$.

A classic result in number theory, Chebyshev's theorem, shows that the fraction of positive integers less than some integer m that are prime is close to $1/(\ln m)$. For example, the fraction of integers less than 10^{100} that are prime is close to $1/(\ln 10^{100}) = 1/230$.

Thus, if we randomly select a large positive odd integer x and sequentially check for primality of the number $x, x+1, x+2, \dots$, That, V on average, this is our first time we'll meet simple number at step number $\ln x$.

4. Equipment And materials

For execution work you will need:

- Class personal computer Not below Pentium-4, cool board, chalk;
- operating room system Microsoft Windows, software tool

L_PROST.EXE.

5. Instructions By technology security

Requirements security before beginning works:

- PROHIBITED: changing clothes, using fire, smoking, eating in the laboratory;
- Ensure that electrical outlets and connectors are intact. Do not operate, break, or attempt to repair faulty power tools.
- V laboratories necessary be V replaceable shoes;
- Turn on the computer only after receiving permission for the work being performed and the permission of the teacher or laboratory assistant.

Requirements security in time works:

- When performing laboratory work, students are required to use only computers, peripheral equipment, connecting cables, measuring equipment and diskettes that are directly related to the given laboratory lesson;

- Connecting and disconnecting components of the computing complex should only be done after all voltage has been completely removed from all components computing complex.

Requirements security By at the end works:

- at the teacher's command, students are required to turn off the computer and components of the computing complex;

- working place bring V order.

Description laboratory work

To perform the lab task, you must run the L_PROST.EXE program . A window appears on the display screen with a text editor located in its center (for displaying information about the actions performed by the program), the main menu is located in the top line of the window, just below the main menu is a toolbar (for managing quick command buttons and other "hot" controls), and at the very bottom of the window, under the text editor, is a status bar, which indicates a hint and displays additional information. For convenience, the toolbar keys are provided with pop-up tips.

To access the main menu, press F10. Navigate through the main menu using the cursor keys. To access a menu item, press "ENTER," and "ESC" to exit the main menu.

For convenience, the program supports mouse operation. To do this, move the pointer to the desired main menu item or toolbar button and press the left mouse button. To cancel, simply press the "ESC" key.

In addition to the main functions of the main menu (“Generate a simple P”, “Search in an interval”, “Check for primality”, etc.), the toolbar contains a key that, when pressed, displays information about the program.

Generation simple P

Possibility of generating a prime number; the number of digits of the generated number is specified by the user (from 1 to 5).

Search in the interval

The ability to search for prime numbers in a given interval. The user specifies: the beginning of the interval - the value x , then the length of the interval - the value L . The search will be performed in the interval $(x, x + L)$.

When checking the primality of each number in an interval, a trial division test is first performed by the first prime numbers in order, followed by a Fermat test. To specify primality testing methods, left-click the checkbox next to the desired method, and then specify all the necessary data to begin the search .

In the trial division method, the initial data is the number of first prime numbers to be divided, while in Fermat's test, the number of bases and their values must be indicated.

Once the calculation is complete, the found prime numbers will be displayed on the screen. and their quantity. A full picture of the work results can be viewed in the menu item “Results output“.

Examination on simplicity

The ability to check the primality of any number. You must enter the number and calculation parameters, similar to searching in an interval.

Output of results

Ability to view all results of the latest processing. When entering the program, before any calculations have been performed, the option to view the source file of the first prime numbers used for the trial division test is available.

Additional intelligence O program

1. When running the prime number generation, range search, and primality checking utilities, the user is asked to confirm that the selected method is correct.
2. In time work long-term By execution procedures The process progress starts and the text editor window is closed. The progress bar allows you to monitor and estimate the approximate speed of the algorithm and the completion time of the current process.
3. Be careful when setting the operating parameters, since during the calculation process these parameters cannot be changed .
4. Description "hot keys":

Ctrl+F1 - 'Generation simple P'

Ctrl+F2 - 'Search in range' Ctrl+F3 -

'Examination on simplicity' Ctrl+F4 -

'Output results' Ctrl+X - 'Exit program'

5. In laboratory work, due to the long counting time, it is recommended to use numbers of no more than five digits and select an interval length of no more than 500, and the number of bases for Fermat's test – no more than 5.

6. *prost.txt* and *work.txt* files must be present in the working directory (along with the *l_prost.exe* file) . It is not recommended to make any changes to these text files, otherwise the consequences may be unpredictable.

6. Tasks

Task No. 1.

1. Check the primality of two arbitrary integers with a digit capacity of at least 5.

2. Distribution simple numbers.

2.1. Given an interval of the form $[x, x+L]$. Calculate the number of prime numbers $\Pi(x, L)$. V interval and compare With the value of $L/\ln(x)$. At Under what conditions is $\Pi(x,L)/L$ close to $1/\ln(x)$?

$x = 2000, L = 500,$

- quantity simple numbers For divisions: 5- 15;

- quantity reasons: 1- 2.

2.2. Find all prime numbers in a given interval. Let $L(i)$ be the difference between two adjacent prime numbers. Plot a histogram for $L(i)$. Calculate the sample mean L_{avg} . Compare with $\ln(x)$.

Where X - middle interval.

Interval: (1000, 1000+300):

- quantity simple numbers For divisions: 5- 20;
- quantity reasons: 1- 3.

2.3. For a given set of numbers $\{ k \}$, estimate the relative error of the formula for *the* k -th prime number:

$$p(k)=k/\ln(k), k = \{10, 15, 20, 30, 35\}.$$

3. Methods generation simple numbers.

3.1. In a given interval, construct a graph of the relative number of natural numbers that pass through the “sieve of Eratosthenes” (i.e. are not divisible by the first k primes).

Interval: (500, 500+200).

Calculation is produced For everyone $k \leq 10$.

3.2. For given interval:

a) calculate the exact amount of P_0 prime numbers in the interval, i.e. when checking, only specify a divisibility test.

The number of first prime numbers to be divided is determined by the calculation: the maximum number to be divided is equal to the square root of the maximum value of the interval.

b) create a test with a small number of trial divisions and one base in the Fermat test. Calculate the number of P_1 probably prime numbers that satisfy this test.

c) create a test with a larger number of trial divisions than in the previous case and two or three bases in the Fermat test. Calculate the number of P_2 likely prime numbers that satisfy this test .

Interval: (1500,1500 + 300).

Analyze received data.

3.3. It is known that there are Carmichael numbers in a given interval.
Find their. Options intervals: (1050, 1050 + 100) (1700, 1700 + 100)
(2400, 2400 + 100)

4. Bring in The report contains answers to test questions, in accordance with the option number specified by the teacher.

7. Content report

Report must contain:

1. Name work, target.
2. Subsequence actions at execution each tasks.
5. Conclusions By execution laboratory work.

8. Control questions

Номер варианта	Контрольные вопросы
1,5,7, 3,9,18,28	Почему в качестве первого основания в тестах типа теста Ферма для проверки на простоту очень больших чисел целесообразно использовать число 2?
2,4,6,8, 20,22,24, 26,30	Какова вероятность $P(x)$ того, что наугад взятое нечетное очень большое число, не превосходящее x , окажется простым?
11,13,15, 10,17,19, 27	Вычислить: $1812 \pmod{13}$, $127 \pmod{7}$.
12,14,16 21,23,25, 29	Сформулируйте суть теста на простоту с использованием пробных делений.

9. A list of recommended literature on this topic

1. Zhuk, A.P., Zhuk, E.P., Lepeshkin, O.M., Timoshkin, A.I. Information Security: A Textbook for Bachelors [Text] / A.P. Zhuk et al. - M.: INFRA-M, 2015. -392 p.
2. Shan'gin, V.F. Computer Information Protection. Effective Methods and Tools [Text] / V.F. Shan'gin – M.: DMK Press, 2012. – 544 p.

Laboratory work No. 6 Electronic digital signature

The purpose of the work: research principles of secure electronic document management in telecommunications networks, algorithms for setting an electronic digital signature (EDS) and the ability to draw independent conclusions based on research results.

2. Formed competencies

This laboratory work is aimed at developing the following competencies of a bachelor's degree graduate:

OPK-1: ability install software And hardware for information and automated systems;

PC-5: ability conjugate hardware And software means as part of information and automated systems.

3. Theoretical Part

Exchanging electronic documents over a communication network significantly reduces the costs of processing and storing documents, and speeds up their retrieval. However, this also poses the problem of authenticating the author of the document and the document itself, i.e., establishing the authenticity of the author and the absence of changes in the received document. When processing documents in electronic form, traditional methods of establishing authenticity based on a handwritten signature and a seal imprint on a paper document are completely unsuitable. A fundamentally new solution is *electronic digital signature (EDS)*. The first EDS scheme, *RSA*, was developed in the late 1970s. However, the problem of authorship verification became relevant.

so much so that it was only in the 1990s, during the explosive growth of the global Internet and the widespread adoption of electronic commerce and services, that a standard was required. It was for this reason that digital signature standards were adopted in Russia and the United States almost simultaneously, in 1994. Of the digital signature schemes proposed by cryptologists, the most successful were *RSA* and the ElGamal scheme. However, the first was patented in the United States and several other countries (the *RSA patent* expired only recently). The second scheme has a large number of possible modifications, and patenting them all is very difficult. Exactly By for this reason scheme EDS El-Gamal remained For the most part, it is patent-free. Furthermore, this scheme has certain practical advantages: the size of the blocks operated by the algorithms, and accordingly the size of the digital signature, are significantly smaller than in *RSA* , while maintaining the same security. This is why the digital signature standards of Russia and the USA are based on the ElGamal scheme. More than 60 countries already have laws on digital signatures. Russia is also on this list. The Law "On Electronic Digital Signature", adopted in 2002, should have a stimulating effect on the development of domestic e-commerce, especially if other regulatory legal acts are promptly brought into line with it.

- The Government of the Russian Federation provides financial support for the implementation of the federal target program “Electronic Russia”.

- The law “On Amendments to Article 80 of Part One of the Tax Code of the Russian Federation” was adopted.

- Back in January 2001, the board of the Pension Fund issued a resolution “On the introduction of cryptographic protection in the system of the Pension Fund of the Russian Federation” information And electronic digital signatures" regulated

registration and connection of legal entities and individuals to their electronic document management system.

- In 2002, the Russian Ministry of Taxes and Duties issued an order “On approval of the procedure for submitting tax returns in electronic form via telecommunications channels connections”, thanks to which Today, any individual or legal entity can contact the tax office, using secure email.

- In 2004, amendments to Articles 13 and 15 of the Law on Accounting were APPROVED BYd, according to which the financial statements of an enterprise can be maintained, stored and submitted to regulatory authorities in electronic form.

The principle of constructing an electronic digital signature. The asymmetry of the roles of sender and recipient in electronic digital signature schemes requires the presence of two closely related keys: *a secret* , or signature key, and *a public* , or signature verification key. Any electronic digital signature scheme must define the following three algorithms:

- algorithm generation key couples For signatures and her checks;
- algorithm productions signatures;
- algorithm checks signatures.

The Russian and US standards are very similar, differing only in some numerical parameters and specific details of key pair generation, signature calculation, and verification. Indeed, both standards are variants of the same ElGamal digital signature scheme. Digital signatures are used to authenticate texts transmitted over telecommunications channels. Functionally, they are similar to a traditional handwritten signature and share its main advantages:

- certifies that the signed text comes from the person who signed it;

- does not give the person himself the opportunity to refuse the obligations associated with the signed text;
- guarantees integrity signed text.

An EDS is a relatively small amount of additional digital information transmitted along with the signed text and includes two procedures:

- a signature procedure that uses the secret key of the message sender;
- a signature verification procedure that uses the sender's public key.

Signature procedure. When generating an EDS, the sender first calculates the hash function $h(M)$ of the text M being signed. The calculated value of the hash function $h(M)$ is one short block information m , characterizing all text M in general. The value m is then encrypted with the sender's secret key. The resulting pair of numbers represents the digital signature for the given text M .

Procedure checks signatures. At verification EDS, recipient messages again calculates hash function $m = h(M)$ accepted By channel text M , after which, using the sender's public key, it checks whether it matches whether received signature calculated meaning m hash functions.

A fundamental point in the digital signature system is the impossibility of counterfeiting a user's digital signature without knowing his secret key.

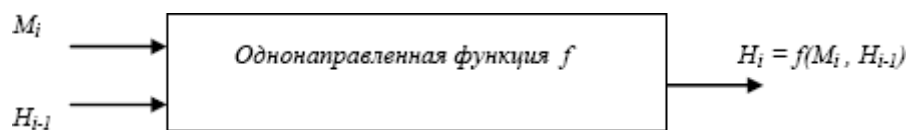
Each one signature, How rule, contains next information:

- date signatures;
- term endings actions key this signatures;
- information O face, signatory text;
- identifier signatory (Name open key);

- actually digital signature.

One-way hash functions. The hash function is designed to compress the document M being signed to several tens or hundreds of bits. The hash function $h(.)$ takes a message M of arbitrary length as an argument and returns a hash value $h(M)=H$ of fixed length. Typically, the hashed information is a compressed binary representation of the underlying message of arbitrary length. It should be noted that the hash value $h(M)$ depends in a complex way on the document M and does not allow the document M itself to be reconstructed. The hash function must satisfy a number of conditions:

- should be sensitive To all sorts of changes V text M ;
- should have property irreversibility, those. task selection to-document M that would have the required hash value must be computationally undecidable;
- the probability that the hash values of two different documents will match should be negligible. Most hash functions is being built on basis unidirectional functions $f(.)$, which produces an output value of length n when given two input values of length n . These inputs are the block of original text M_i and the hash value H_{i-1} of the previous block of text, Fig. 1.



Drawing 1. Formation hash functions

Algorithm digital signatures DSA. Algorithm digital signatures *DSA (Digital Signature Authorization)* proposed V 1991 year V USA And is-

The development of the ElGamal digital signature algorithm is being pursued. The sender and recipient of an electronic document use large integers in the calculation:

G, P - simple numbers By L -bit each ($L = 512 \dots 1024$ bit); q is a prime number of length 160 bits divisor of the number $(P - 1)$.

G, P, q are open and can be shared by all network users.

Description algorithm

1. The sender selects a random integer X , $1 < X < q$. The number X is the sender's *secret key* for generating the electronic signature.

2. Sender calculates meaning

$$Y = G^X \bmod P.$$

The number Y is *the public key* for verifying the sender's signature and is transmitted to all recipients of the document.

3. In order to sign a document M , the sender hashes it into an integer hash value m :

$$m = h(M), 1 < m < q.$$

Then generates random whole number K , $1 < K < q$ And calculates number-
to r :

$$r = (G^K \bmod P) \bmod q.$$

4. At help secret key X sender calculates number s :

$$s = ((m + r \cdot X) / K) \bmod q.$$

Pair numbers r, s form digital signature $S = (r, s)$ under document-volume M .

5. The message delivered to the recipient, together with the signature, is a triple of numbers $[M, r, s]$. First of all, the recipient checks that the following relations are met:

$$0 < r < q; 0 < s < q.$$

6. Next recipient calculates values:

$$w = 1/s \bmod q;$$

$$m = h(M) - \text{hash value}; u_1 =$$

$$(m \cdot w) \bmod q;$$

$$\text{and } u_2 = (r \cdot w) \bmod q.$$

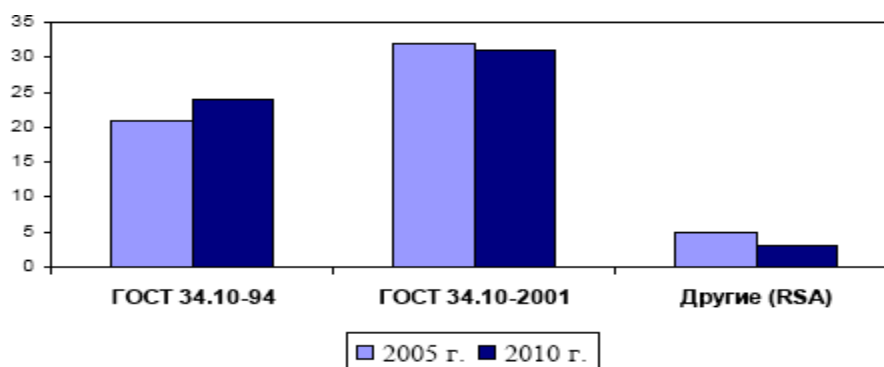
Then at help open key Y is calculated meaning

$$v = ((G^{u_1} \cdot Y^{u_2}) \bmod P) \bmod q,$$

and the equality $v = r$ is checked. If it is satisfied, then the signature is considered authentic, since it can be strictly mathematically proven that the last equality will be satisfied if and only if signature $S = (r, s)$ under document M received at help exactly of the secret key X from which the public key Y was obtained.

New standards EDS. Latest achievements cryptography showed that the general problem of logarithm taking in discrete fields, which is the basis of the indicated digital signature scheme, cannot be considered a sufficiently solid foundation. For example, the block sizes considered "safe" are growing at a comparatively rapid pace. As a result, the digital signature standards of Russia and the USA were updated in 2001: they were transferred to elliptic curves. The digital signature schemes remained the same, but the numbers they operate on are now not elements of the finite field $GF(2n)$ or $GF(p)$, but elliptic numbers - solutions equations elliptical curves over specified final By-

lamia. The role of the operation of raising a number to a power in a finite field in the updated standards is performed by the operation of taking a multiple point of an elliptic curve – “multiplication” of the point by an integer.



Drawing 2. Algorithms EDS, used in the Russian Federation

By properly selecting the elliptic curve type, it becomes significantly more difficult to crack an electronic digital signature scheme and reduces the working size of data blocks. The old Russian electronic digital signature standard operates with 1024-bit blocks, while the new one, based on elliptic curves, operates with 256-bit blocks, yet offers greater security. The security of the GOST R34.10-94 signature scheme is based on the complexity of solving the discrete logarithm problem in a prime field. Currently, the fastest algorithm for solving it for the general case is the generalized sieve algorithm for a number field. In GOST R34.10-2001, the security of the electronic digital signature scheme is based on the complexity of solving the discrete logarithm problem in a group of points on an elliptic curve. With the correct choice of curve parameters, the most effective methods for solving it are the more labor-intensive Pollard *r*- and *l* -methods. Thus, according to various expert estimates, the complexity of hacking the old and new Russian digital signature standards is approximately 1026 and 1038 multiplication operations in the base field $GF(p)$, respectively. Obviously, the new standard is more secure.

4. Equipment And materials

For execution work you will need:

- Class personal computer Not below Pentium-4, cool board, chalk;
- operating room system Microsoft Windows, software tool
labWork6.exe

5. Instructions By technology security

Requirements security before beginning works:

- PROHIBITED: changing clothes, using fire, smoking, eating in the laboratory;
- Ensure that electrical outlets and connectors are intact. Do not operate, break, or attempt to repair faulty power tools.
- V laboratories necessary be V replaceable shoes;
- Turn on the computer only after receiving permission for the work being performed and the permission of the teacher or laboratory assistant.

Requirements security in time works:

- When performing laboratory work, students are required to use only computers, peripheral equipment, connecting cables, measuring equipment and diskettes, directly related To given laboratory occupation;
- connection and disconnection of components of the computing complex should be carried out only with complete removal of tension from all components computing complex.

Requirements security By at the end works:

- at the teacher's command , students are required to turn off the computer and components of the computing complex;
- working place bring V order.

6. Tasks

Task No. 1. Familiarize yourself with the main areas of work within the framework of the federal target program “Electronic Russia”, as well as with information on the procedure for using and the current algorithms for setting up an electronic digital signature , *set out in paragraph 3. Run the labWork6.exe* program , designed to demonstrate the procedure for setting up and verifying an electronic digital signature.

Task #2. Generate and send keys for encrypting the original document and keys for signing the document to the exchange participants. The original text for encryption is typed directly in the program window .

Task #3. Encrypt the original message and sign it using the sender's secret key.

Task #4. Forward an encrypted and signed message to the recipient. Verify the digital signature and restore the original message text.

Task No. 5. Save in the report screen forms demonstrating the process of key generation and distribution; the process of encrypting the original document and setting the digital signature.

Task #5. Provide answers to the test questions in your report, according to the option number provided by the teacher.

7. Content report

Report must contain:

1. Name work, target.
2. Subsequence actions at execution each tasks.
3. Conclusions By execution laboratory work.

8. Control questions

Номер варианта	Контрольные вопросы
1,5,7, 3,9,18,28	В чем состоит назначение хэш-функций и какие требования предъявляются к хэш-функциям, используемым для постановки ЭЦП? Перечислите стандарты хэш-функций, действующие в Российской Федерации.
2,4,6,8, 20,22,24, 26,30	Опишите процедуры постановки и проверки ЭЦП. Какая информация содержится в ЭЦП?
11,13,15, 10,17,19, 27	Перечислите стандарты ЭЦП, действующие в Российской Федерации. На каких принципах основана криптостойкость современных алгоритмов ЭЦП?
12,14,16 21,23,25, 29	Приведите пример реализации алгоритма ЭЦП (<i>RSA, El-Gamal, DSA</i>)

9. A list of recommended literature on this topic

1. Zhuk, A.P., Zhuk, E.P., Lepeshkin, O.M., Timoshkin, A.I. Information Security: A Textbook for Bachelors [Text] / A.P. Zhuk et al. - M.: INFRA-M, 2015. -392 p.
2. Shan'gin, V.F. Computer Information Protection. Effective Methods and Tools [Text] / V.F. Shan'gin – M.: DMK Press, 2012. – 544 p.

Lab #7 Encryption method sliding permutations

The purpose of this work is to study the sliding transposition cipher using *the XY-Move software implementation* and to develop the ability to draw independent conclusions based on the research results.

2. Formed competencies

This laboratory work is aimed at developing the following competencies of a bachelor's degree graduate:

OPK-1: ability install software And hardware for information and automated systems;

PC-5: ability conjugate hardware And software means as part of information and automated systems.

3. Theoretical Part

Persistent Plaintext Patterns and Their Use in Deciphering Simple Substitution and Transposition Ciphers

Opportunity decryption any cipher V significant depends on the extent to which cryptographic transformations destroy the probabilistic-statistical regularities present in the plaintext. Thus, in meaningful texts of any natural language, different letters occur with different frequencies, while the relative frequencies of letters in different texts of the same language are close to each other. The same can be said about the frequencies of pairs and triplets of letters of the plaintext. text. Except Togo, any natural language possesses So

called redundancy, which allows one to “guess” the meaning of a message with a high probability, even if some of the letters in the message are unknown.

Table 1. Relative frequencies appearances letters alphabet Russian language in the text

1	а - 0,062	12	л - 0,035	23	ц - 0,004
2	б - 0,014	13	м - 0,026	24	ч - 0,012
3	в - 0,038	14	н - 0,053	25	ш - 0,006
4	г - 0,013	15	о - 0,090	26	щ - 0,003
5	д - 0,025	16	п - 0,023	27	ы - 0,016
6	е,е - 0,072	17	р - 0,040	28	ь,ь - 0,014
7	ж - 0,077	18	с - 0,045	29	э - 0,003
8	з - 0,016	19	т - 0,053	30	ю - 0,006
9	и - 0,062	20	у - 0,021	31	я - 0,018
10	й - 0,010	21	ф - 0,002	32	- 0,175
11	к - 0,028	22	х - 0,009		

Similar tables are given in various books. They are obtained based on frequency calculations on large volumes of plaintext. Considering that different source material is used for experiments, probability values some differ among themselves. If we arrange the letters in descending order of probability, we will obtain the following variational series:

O, E, A, I, N, T, S, R, V, L, K, M, D, P, U, Ya, Z, Y, B, Ъ, G, CH, Y, X, Zh, Yu, Sh, Ts, Shch,

E,F

For example, the word **СЕНОБАЛИТ** contains the 10 most frequently occurring letters.

The frequencies of alphabetic characters depend not only on the language, but also on the nature of the text. For example, in a text on cryptography, the probability of the letters F, SH will be increased (due to the frequent occurrence of the words “cipher”, “cryptography”). In some mathematical texts, the frequency of the letter F may be overestimated (due to the words “function”,

“functional”, etc.). In standard text files, the most frequent is symbol "space".

Frequency content diagram

of random texts is a stable characteristic of the text. From probability theory it follows that under sufficiently weak restrictions on the probabilistic properties of a random process, the law of large numbers is valid, i.e., the relative frequencies g/N of characters converge in probability to the values of their probabilities p_k

$$P\left\{\left|\frac{g_k}{N} - p_k\right| > \varepsilon\right\} \xrightarrow{N \rightarrow \infty} 0.$$

Transposition and simple substitution ciphers are not completely destroy the probabilistic-statistical properties present in the open message.

At decryption text, encrypted in cipher simple substitutions, use the frequency characteristics of the plaintext. Specifically, if we count the frequencies of occurrence of characters in the ciphertext, arrange them in descending order, and compare them with the variational series of probabilities of the plaintext, then these two sequences will be close. Rather, total on first place it will turn out space, further will be next vat letters ABOUT, E, A, AND.

Of course, if the text isn't very long, it doesn't necessarily have to be a complete match. An O might be in second place, and an E in third, but in any case, identical letters in the first and second rows will be located close to each other, and the closer to the beginning (the greater the probability of the characters), the smaller the distance between the characters.

A similar picture is observed for pairs of adjacent letters (bigrams) of the plaintext (the most frequent bigram of the Russian plaintext is ST). However, to obtain a stable picture, the length of the analyzed sequence must be large enough. On relatively small segments of plaintext, this picture is somehow blurred. A more stable characteristic bigram is absence V meaningful text some bigram, How They say, availability forbidden bigram, name-

with a probability equal to practically 0. Have you ever seen in plaintext the bigram $\mathbb{B}\mathbb{B}$, or bigrams of the form: “vowel” $\mathbb{B}$; “space” $\mathbb{B}$? Knowing and using these features of plaintext significantly facilitates deciphering a transposition and substitution cipher.

Permutation cipher . Let X be the set of open (meaningful) texts in alphabet I . The lengths of all possible open texts are multiples of T . The set of keys is the symmetric permutation group S_T of degree T , for $g \in S_T$ we define the encryption function fg by setting for $(i_1, i_2, \dots, i_T) \in X$

$$fg(i_1, i_2, \dots, i_T) = (i_{g(1)}, i_{g(2)}, \dots, i_{g(T)});$$

we define fg on the remaining elements of X according to the rule: the text $x \in X$ is divided into segments of length T and each segment of length T is encrypted with the key g according to the encryption law specified above. The sequence composed of letters of the images of encrypted words is the ciphertext corresponding to the plaintext x and the key g . To encrypt a text of length not a multiple of T , it is supplemented with letters to a length multiple of T .

Decrypting a transposition cipher. The ciphertext is written into a table with T columns. To recover the plaintext of the transposition cipher, we need to rearrange the columns so that meaningful text appears in the rows .

Let's consider an example of decrypting an eight-column transposition cipher (Textbook "Principles and Methods of Information Security" by G.V. Proskurin). Let the ciphertext have the following form:

1	2	3	4	5	6	7	8
п	а	я		в		н	м
о	ч	ш	г		у		е
е	б	ж	л		е		о
м		ч		о	т	о	я
	е	г	е		у	с	ш
	а	к	ь	з	а	т	т
я	р	е		е	п		ь
	ю	з	в	а	н	в	
о	й	а	в	е	ш	л	
	е	е	я	м		п	н
ь	р	р	н	з	е	е	е
з	а	м	а	н		а	к
ч	с	т	а		ь	а	н
о	я	л	м		а	л	
о	ь	ч	х	т	а	т	
в		е	о	а	л	е	п
о	е	р	м	т	ь	е	
д	с	г	ы		о	а	т
е	б	в	н		ы		
	а	у	и	н	з	н	л
	г	и	а	о	к	к	д
	а	о	б	д	г	н	
	ж	а	у	е	д	я	д
х	л	и		е	м	о	а
к	р	т	д		ь	о	е
	ь	х	в	т	о	н	
р	л	е		е	д	а	ю
р		з	е	в		е	д
ш		в	а	е	н	е	н
т	и	й	е	в			д
		в		с	д		

Let's associate a permutation of columns with an 8*8 table, and put a one at the intersection of the i-th row and the j-th column if the j-th column, after the inverse permutation, should follow the i-th. Our task is to restore the table corresponding to the correct permutation of columns.

Let's now pairwise add one column to another. If forbidden bigrams appear in some rows, then the columns cannot follow each other in the plaintext and the corresponding cell is crossed out. In our example, the 6th column cannot follow for 4th, because otherwise V text V first line will contract two sample-

1a. Let's look at the 6th row, for example. If the 4th column followed the 1st, the text would contain words beginning with B.

After viewing everyone lines We we will receive next table:

	1	2	3	4	5	6	7	8
1	x	x		x	x	x		x
2		x		x		x		
3			x					x
4	x	x		x		x	x	x
5	x				x	x	x	x
6	x	x		x		x	x	
7	x			x	x	x	x	x
8	x	x			x	x	x	x

If would text was longer And there were lines would more, That V each row and each column would have exactly one uncrossed-out cell left and the permutation would have been restored.

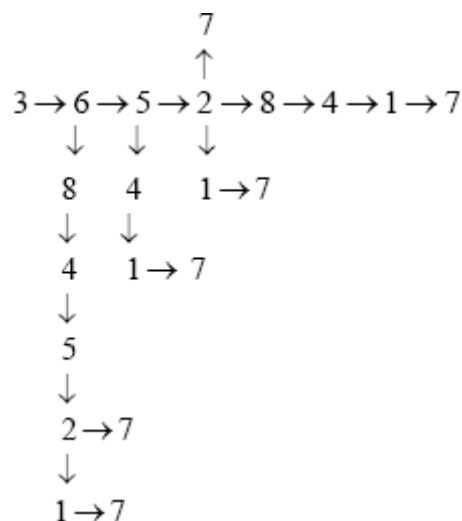
In our table, we can only assert that the 6th column follows the 3rd (let's denote this event as $3 \rightarrow 6$) if the 6th column is not the last one. There are two possible continuation options for the 6th column:

$$\begin{array}{c} 8 \\ \uparrow \\ 3 \rightarrow 6 \rightarrow 5 \end{array}$$

We need to consider both and try to eliminate the false option. If we can't eliminate the false option, then we need to continue with both options.

$$\begin{array}{ccccc} 8 \rightarrow 4 & & 1 & & \\ \uparrow & & \uparrow & & \\ 3 \rightarrow 6 \rightarrow 5 \rightarrow 2 \rightarrow 7 & & & & \end{array}$$

As a result, we get some tree of possible sequence of columns in plain text.



Each branch of the tree corresponds to a certain permutation of columns. Next, we check each option for meaningfulness and obtain the correct one.

ny option.

3 → 6 → 5 → 2 → 8 → 4 → 1 → 7

Note that it wasn't necessary to build the tree all the way to the end. For example, the branch 3 → 6 → 8 → 4 → 5 could have been eliminated right away. How could the following fragment of text be considered meaningful?

3	6	8	4	5
я		м		в
ш	у	е	г	
ж	е	о	л	
ч	т	я		о
г	у	щ	е	з
к	а	т	ь	е
е	а	т		а

Such a procedure of pruning branches would be simply necessary if there were fewer rows, and the tree would be correspondingly much more branched.

The proposed procedure is easy to automate and suitable for computer implementation. The decryption algorithm should consist of the following steps.

1. Preliminary work. By analyzing a sufficiently representative volume of plaintexts, construct a set of forbidden bigrams.
2. Preliminary work. Compile a dictionary of all possible v -grams for $v=2,3,\dots,d$ that may appear in plaintext. The number d is chosen based on the capabilities of the computing technology.

Construct an 8×8 table. All forbidden bigrams are tried sequentially, and for each bigram being tried, all rows are tried sequentially. If in at least one row the first symbol of the bigram occurs in the i th column, and the second in the j th, then the $i \times j$ cell of the table is crossed out.

3. Choose some column V quality initial column.
4. Start the tree construction procedure by appending all column variants to the original column.
5. For each resulting variant, add one more of the remaining columns. If at least one row of the table contains a 3-gram that is not in the dictionary of available 3-grams, the variant is eliminated.
6. For each of the unfiltered variants, we add another column and filter out false variants using the dictionary of permitted 4-grams.

If the dictionary was built only for $d \leq 3$, then filtering is performed by checking the admissibility of the 3-grams occurring in the last three columns of each row. We continue this process until a complete permutation is obtained.

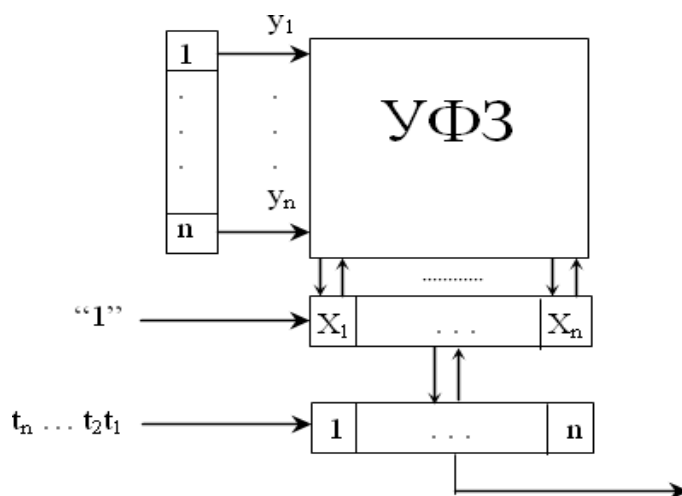
Below, V table, given restored For ours example text.

	1	2	3	4	5	6	7	8
1	я		в	а	м		п	и
2	ш	у		ч	е	г	о	
3	ж	е		б	о	л	е	
4	ч	т	о		я		м	о
5	г	у		е	ш	е		с
6	к	а	з	а	т	ь		т
7	е	п	е	р	ь		я	
8	з	н	а	ю		в		в
9	а	ш	е	й		в	о	л
10	е		м	е	н	я		п
11	р	е	з	р	е	н	ь	е
12	м		н	а	к	а	з	а
13	т	ь		с	н	а	ч	а
14	л	а		я		м	о	л
15	ч	а	т	ь		х	о	т
16	е	л	а		п	о	в	е
17	р	ь	т	е		м	о	е
18	г	о		с	т	ы	д	а
19	в	ы		б		н	е	
20	у	з	н	а	л	н		н
21	е	к	о	г	д	а		к
22	о	г	д	а		б		н
23	а	д	е	ж	д	у		я
24	н	м	е	л	а		х	о
25	т	ь		р	е	д	к	о
26	х	о	т	ь		в		н
27	е	д	е	л	ю		р	а
28	з		в		д	е	р	е
29	в	н	е		н	а	ш	е
30	й		в	и	д	е	т	ь
31	в	а	с					

Transposition ciphers were further developed by implementing the idea of continuous local permutation of plaintext symbols under the action of a control sequence. To implement the mixing of plaintext symbols, individual text symbols are stored in the encoder's memory and their transmission is delayed in discrete time. We introduce the parameters n_1 and n_2 such that $n = n_1 + n_2$. In these ciphers, the i -th symbol of the transmitted message is rearranged in the encrypted message to the j -th place, where $n_1 \leq j \leq i + n_2$. The control sequence of the delay time is chosen so that the delay time of each symbol is a random variable with a uniform distribution, that is, the probability of each fixed value of the delay time should be close to $1/n$.

Encrypting machine sliding permutations

Let us consider the circuit diagram of an encryption machine that, with a suitable control sequence, allows the implementation of an arbitrary sliding permutation shi, Fig. 7.1.



Rice. 7.1. Scheme implementation encryption machine gun sliding permutations at the entrance formation node delays (UFZ) in each tact t is served vector

Knot formations delays is final auto ohm m ($F^n, F^n, \{1, \dots, n\}, \delta$, λ), the set of states of which is the set all possible binary vectors - fillings $x(t) = (x_1^t, \dots, x_n^t)$ lower accumulator.

IN tact t is being produced natural number - meaning delays.

$$\gamma_t = \max_j j : \left\{ x_i^t \mid y_i^t \in \{0,1\}, i = \overline{2, n-1}, y_1^t = y_n^t = 1. \right.$$

At this machine passes V following state:

$$\vec{x} = (t+1) = (x_1^{t+1}, \dots, x_n^{t+1}),$$

где

$$x_1^{t+1} = 1,$$

$$x_i^{t+1} = \begin{cases} 0, & i = \gamma_t + 1 \\ x_{i-1}^t, & i \in \{2, \dots, \gamma_t, \gamma_t + 2, \dots, n\} \end{cases}$$

Signs plaintext is recorded to the bottom storage device. In line in *the* t -th clock cycle, the plaintext character written in the cell with is sent number.

States machine gun $x(t)$ are indicators, showing which of the plaintext characters have not yet been read from the lower storage device.

To encrypt the sequence $t_N, \dots, t_2, t_1$ proceed as follows. First, we write the first $n-1$'s into the lower accumulator plaintext characters

$$(t_{n-1}, \dots, t_1, 0, \dots, 0)$$

Simultaneously machine is being installed V initial state

$$X(1) = (0, \dots, 0).$$

After this, the UFZ machine begins to operate according to the law described above until By V storage device Not will arrive last sign t_N plaintext. When the supply of plaintext characters to the accumulator ceases, the supply of units to the indicator accumulator ceases. In the remaining $n-1$ clock cycles, the information recorded in the storage device is read.

When decrypting, the UFZ works according to the same scheme, only instead of reading, it is necessary to organize the recording of information into the second storage device.

Особенности работы УФЗ

В каждом такте t (за исключением последних n_1 тактов) в накопитель-индикаторе $\vec{x}(t)$ записано ровно n_1 единиц. Поэтому в такте t величина задержки может принимать только одно из n_1 значений в интервале $\{1, \dots, n\}$. В частном случае, когда $n=1$, либо $n_1=n$ УФЗ вырабатывает постоянно значения задержки $\gamma_t = 1$ и $\gamma_t = n$ соответственно. Легко видеть, что результирующее преобразование открытого текста действительно будет шифром скользящей перестановки. Условие $y_1^t = 1, t=1, 2, \dots$ обеспечивает постоянное считывание во всех тактах, а условие $y_2^t = 1$ ограничивает величину задержки $\gamma_t \leq n$.

Example, explanatory work encryption machine gun sliding permutations

Let's take For example $n=7, n_1=3, n_2=4$

On entrance Federal Law No. on each step t work encryption machine gun vector is supplied

$\vec{y} = (y_1^t, \dots, y_n^t)$, V linear register shift (LRS), rice. 7.2.

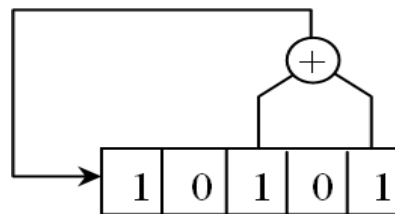


Fig.7.2. Linear register shift

We hope that the reader will be able to write the sequence of states of this linear shift register, which forms the control sequence of the encryption machine.

We will denote at each step of the operation of the encryption machine the sequence $Y_1, Y_2, \dots, Y_n$ (V ours case $Y_1, Y_2, \dots, Y_7$), incoming With LRS, How

"Y", and the sequence of units $X_1, X_2, \dots, X_n$ (in our case $X_1, X_2, \dots, X_7$) as "1". In the bottom line we will write down the characters of the plaintext that are at this step in the lower accumulator of the encryption machine $t_1, t_2, \dots, t_n$ (in our case $t_1, t_2, \dots, t_7$). Let's look at the work step by step encoder under specific conditions.

1-й шаг

"Y"	1	1	0	1	0	1	1
"1"	1	1	1	0	0	0	0
	t_3	t_2	t_1	0	0	0	0

2-й шаг

"Y"	1	0	1	0	1	0	1
"1"	1	1	0	1	0	0	0
	t_4	t_3	t_2	t_1	0	0	0

3-й шаг

"Y"	1	0	0	1	0	1	1
"1"	1	0	1	0	1	0	0
	t_5	t_4	t_3	t_2	t_1	0	0

4-й шаг

"Y"	1	0	0	0	1	0	1
"1"	1	0	0	1	0	1	0
	t_6	t_5	t_4	t_3	t_2	t_1	0

5-й шаг

"Y"	1	0	0	0	0	1	1
"1"	1	0	0	0	1	0	1
	t_7	t_6	t_5	t_4	t_3	t_2	t_1

6-й шаг

"Y"	1	1	0	0	0	0	1
"1"	1	1	0	0	0	1	0
	t_8	t_7	t_6	t_5	t_4	t_3	t_2

7-й шаг

"Y"	1	0	1	0	0	0	1
"1"	0	1	0	0	0	0	1
	0	t_8	t_7	t_6	t_5	t_4	t_3

8-й шаг

"Y"	1	0	0	1	0	0	1
"1"	0	0	1	0	0	0	0
	0	0	t_8	t_7	t_6	t_5	t_4

At each step we searched starting from the left edge and going to the right for the first match V lines "Y" and "1" (1 V both lines) And For facilities outlined this column. The plaintext element that ended up in the selected column goes away V line. Like this in this way, V ours example, subsequence the one that went into line has the following form: $t_2, t_4, t_5, t_6, t_1, t_7, t_3, t_8$

Description software implementation

To complete the laboratory work, you must install the XY-Mover software module on your computer.

1) The menu bar in this program consists of three items: Encryption, View, Help. The desired menu item can be selected by clicking on it. left mouse button, or with using the right arrow keys on the keyboard,

"Left," after pressing the "F10" function key. Once the user has selected the desired menu item, a drop-down submenu will open. Let's take a closer look at the menu items.

- *Encryption.* Drop-down menu items can be selected either with the left mouse button or with the up and down keys on the keyboard. Let's take a closer look at the submenu items.

- Editing parameters – this menu function allows you to set the required parameters in the “Encoder parameters” field of the program window;

- Encrypt - choice this point menu launches process encryption;

- Decryption – selecting this menu item starts the decryption process ;

- Exit - completion work programs.

- *View.* This menu item allows you to select the program's appearance. The submenu items are listed below:

- Parameters - choice this point menu allows use field

"Parameters encoder", described below.

- Scheme encoder - choice this point menu allows observe structural diagram of the encryption machine.

- Status Bar – Selecting this menu item allows you to view the status bar, described below.

2) Panel tools

The user may find it more convenient to use the toolbar (called "buttons") instead of the menu. "Buttons" duplicate some menu items, but selecting a "button" is much more convenient than selecting a submenu item. To do this, simply left-click on the desired button. Let's look at the "buttons" from left to right:

- "A", choice this buttons allows get result similar to the menu item: View-Options;
- "Encoder Scheme" – selecting this button allows you to get a result similar to the menu item View - Encoder Scheme;
- "Status Bar" – selecting this button allows you to get a result similar to the menu item: View - Status Bar;
- "Edit parameters" – selecting this button allows you to get a result similar to the menu item: Encryption-Edit parameters;
- "Encrypt" – selecting this button allows you to get a result similar to the menu item Encryption – Encrypt;
- "Decrypt" – selecting this button allows you to get a result similar to the menu item Encryption – Decrypt;
- "Abort" – selecting this button allows you to interrupt the data processing process (encryption or decryption).

3) Line states

The status bar is located at the bottom of the program window. It displays information about the program's status:

- "Sliding Permutation Encryption" - This message indicates that the program is ready to run.
- "Completed ... %»" - indication volume completed work at encryption or decryption.

4) Description fields windows programs

- Parameters encoder

- *n1* - number signs, which are being recorded V lower storage device first;
- *n2* - rest signs (total their 127);
- bends register - points removal LRS;
- initial filling - initial filling LRS, which The user can change.

- Input flow

- a field that displays the name of the text file whose contents need to be encrypted/decrypted;

- "Open" button - when you click this button, the standard *Windows* "Open File" window opens;

- text - content file, open For encryption/decryption;
- bit flow - bitwise performance symbols selected file.

- Day off flow

- a field that displays the name of the text file whose contents need to be decrypted/encrypted;

- "Open" button - when you click this button, the standard *Windows* "Open File" window opens;

- text - content file, open For decryption/encryption;
- bit flow - bitwise performance symbols selected file.

4. Equipment And materials

For execution work you will need:

- Class personal computer Not below Pentium-4, cool board, chalk;
- Microsoft Windows operating system , software *labWork6.exe*

5. Instructions By technology security

Requirements security before beginning works:

- PROHIBITED: changing clothes, using fire, smoking, eating in the laboratory;
- Ensure that electrical outlets and connectors are intact. Do not operate, break, or attempt to repair faulty power tools.
- V laboratories necessary be V replaceable shoes;
- Turn on the computer only after receiving permission for the work being performed and permission from the teacher or laboratory assistant.

Requirements security in time works:

- When performing laboratory work, students are required to use only computers, peripheral equipment, connecting cables, measuring equipment And floppy disks, directly related to this laboratory lesson;
- connection and disconnection of components of the computing complex should be carried out only with complete removal of tension from all components computing complex.

Requirements security By at the end works:

- at the teacher's command, students are required to turn off the computer and components of the computing complex;
- working place bring V order.

6. Tasks

1. For execution laboratory work on computer necessary Install the XY-Mover software module.
2. Execute initial installations encoder, according to example.
3. Download file For encryption.
4. Produce encryption information With using cipher sliding permutation, save the ciphertext to a file.

5. Describe in the report the process of encryption and decryption of data using the XY-Mover emulator program. Analyze the obtained data.

6. Include in the laboratory report the answers to the control questions selected in accordance with the option number specified by the teacher.

7. Content report

Report must contain:

1. Name work, target.
2. Subsequence actions at execution each tasks.
3. Conclusions By execution laboratory work.

8. Control questions

Номер варианта	Контрольные вопросы
1,5,7, 3,9,18,28	Почему шифрование методом гаммирования является наиболее подходящим для высокоскоростных линий телекоммуникационной связи?
2,4,6,8, 20,22,24, 26,30	Какие общие требования, предъявляются к гамме шифра?
11,13,15, 10,17,19, 27	Приведите пример, поясняющий работу шифрующего автомата скользящей перестановки при $n=5$, $n1=2$, $n2=3$.
12,14,16 21,23,25, 29	Кратко опишите работу схемы реализации шифра скользящей перестановки.

9. List literature, recommended To use By this topic

1. Zhuk, A.P., Zhuk, E.P., Lepeshkin, O.M., Timoshkin, A.I. Information Security: A Textbook for Bachelors [Text] / A.P. Zhuk et al. - M.: INFRA-M, 2015. - 392 p.

2. Shan'gin, V.F. Computer information protection. Effective methods and tools [Text] / V.F. Shan'gin – M.: DMK Press, 2012. – 544 p.

LABORATORY JOB No. 8

STUDYING SOFTWARE PRODUCTS ZI.

PROGRAM *PGP* (Pretty Good Privacy)

Objective of the work: To become familiar with the general principles of constructing and using software tools for information security, in particular with the **PGP program**.

Development funds software systems **PGP** , intended For:

- encryption confidential resources And demarcations access To him;

- provision integrity informational resources With help fur-Nism electronic digital signatures;

- reliable destruction residual confidential information;
- hiding presence V computer system confidential information using a virtual disk.

- For execution laboratory work at absence on On your computer, you need to install the PGP program.

- Installation file attached To description laboratory PGPfreeware602i. Select only the following components for installation:

- PGP 6.0.2 Program Files;
- PGP 6.0.2 User's Manual;
- Unconfigured PGP 6.0.2 Client Install;
- PGPdisk for Windows.

When the installation program asks about the existence of keys, answer “No”, and when it asks about the need to reboot the system, answer “Yes”.

Description laboratory *PGP 6.0*

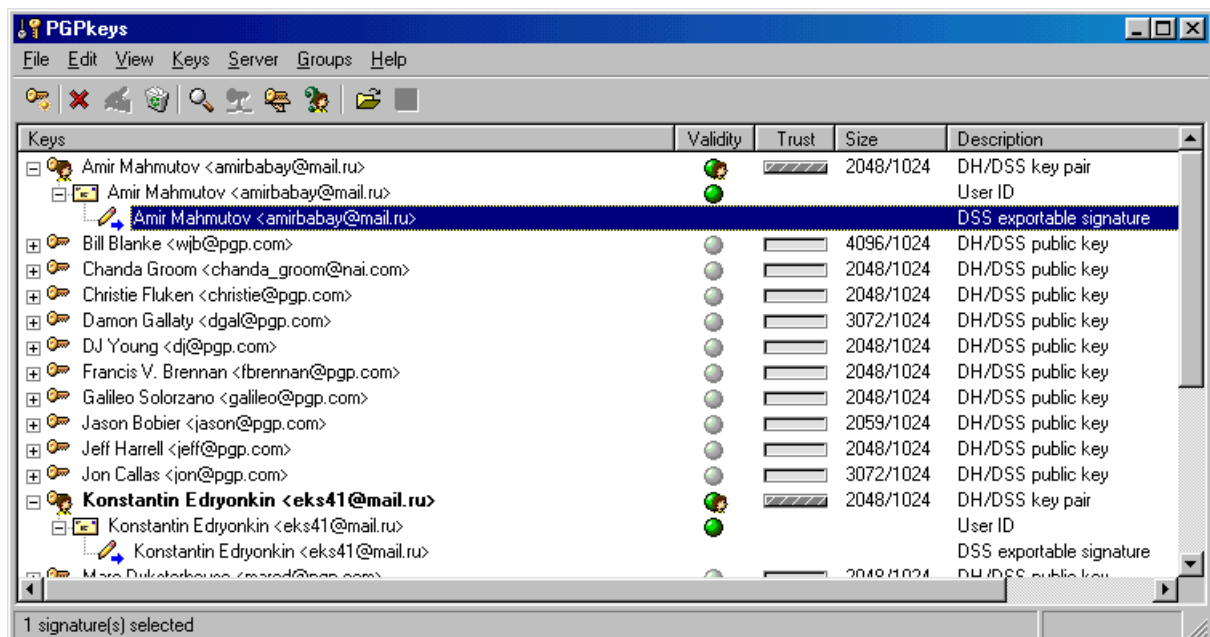
Features

The primary functions of *PGP* are to encrypt files and email messages, certify them with electronic signatures, and completely destroy files on disk. Additionally, *PGP* provides the following features:

- storage open keys on remote server;
- usage three symmetrical algorithms encryption And two asymmetrical;
- four launch methods : *E-mail plugins*, *PGP tray*, *PGP tools* , context menu;
- separation keys;
- installation level validity key And trust to the owner key.

PGPkeys

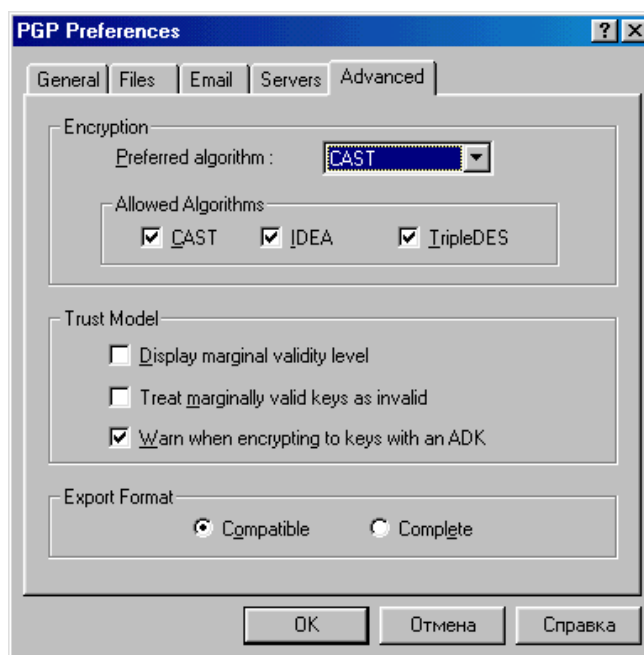
This is a program included in *PGP 6.0* and provides work with keys, Fig. 8.1.



Rice. 8.1 PGPkeys window

The **File** menu contains two commands, one is **Exit** , the other is **Send Key Shares** which allows you to send a local copy of the shared key over the network.

Menu **Edit** except ordinary teams contains **Preferences** command , which is used to set program settings, Fig. 8.2.



Rice. 8.2. PGPPreferences window

Using the **View menu** , you can set the properties of the keys that are displayed on the screen .

Teams menu *Keys*

Sign – allows you to sign with your private key other users' public keys. This way, you can show that you trust the owner of the key being used, that is, the key really belongs to the owner. You can specify modifiers: 1) *Non-Exportable* - for a local set of keys; 2) *Exportable* - the certified key is sent to the server; 3) *Meta-introducer Non-Exportable* - you trust not only the owner, but also to any trusted person them key; 4) *Trusted Introducer Exportable* - You do you trust to the owner sign keys, Fig.8.3.

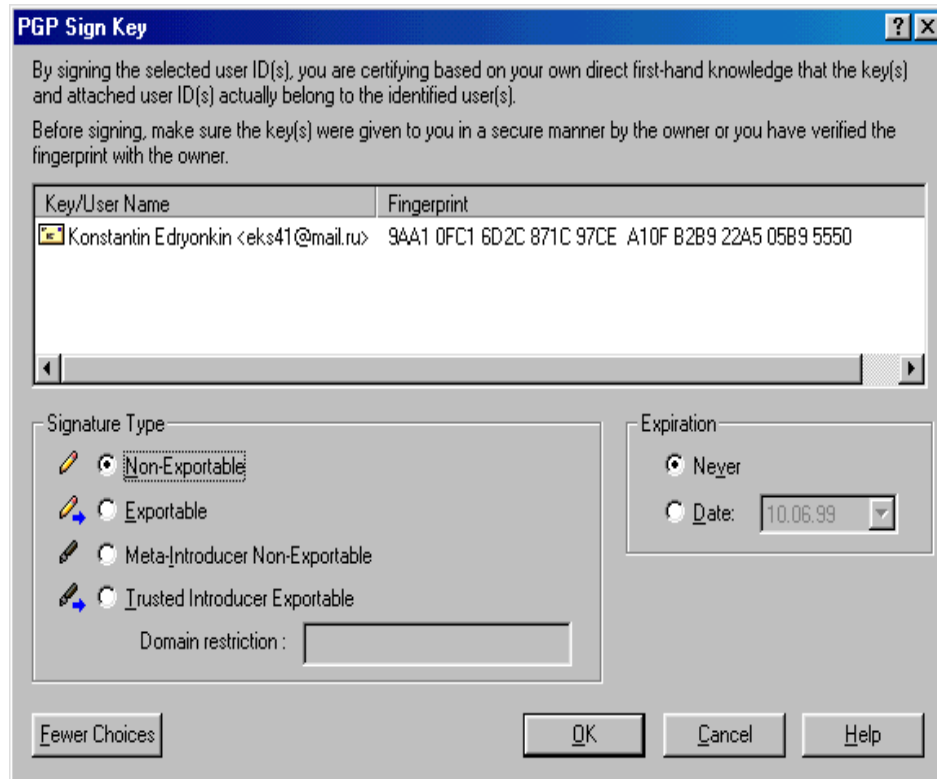


Fig.8.3. Window PGP SignKey

Set as Default Key – designates this key as the default. **Add Name/Photo/Revoker** – allows you to add the name of the owner with the right to invalidate your key to the key properties.

Revoke - announce key invalid.

Reverify Signatures - check signature (right) key.

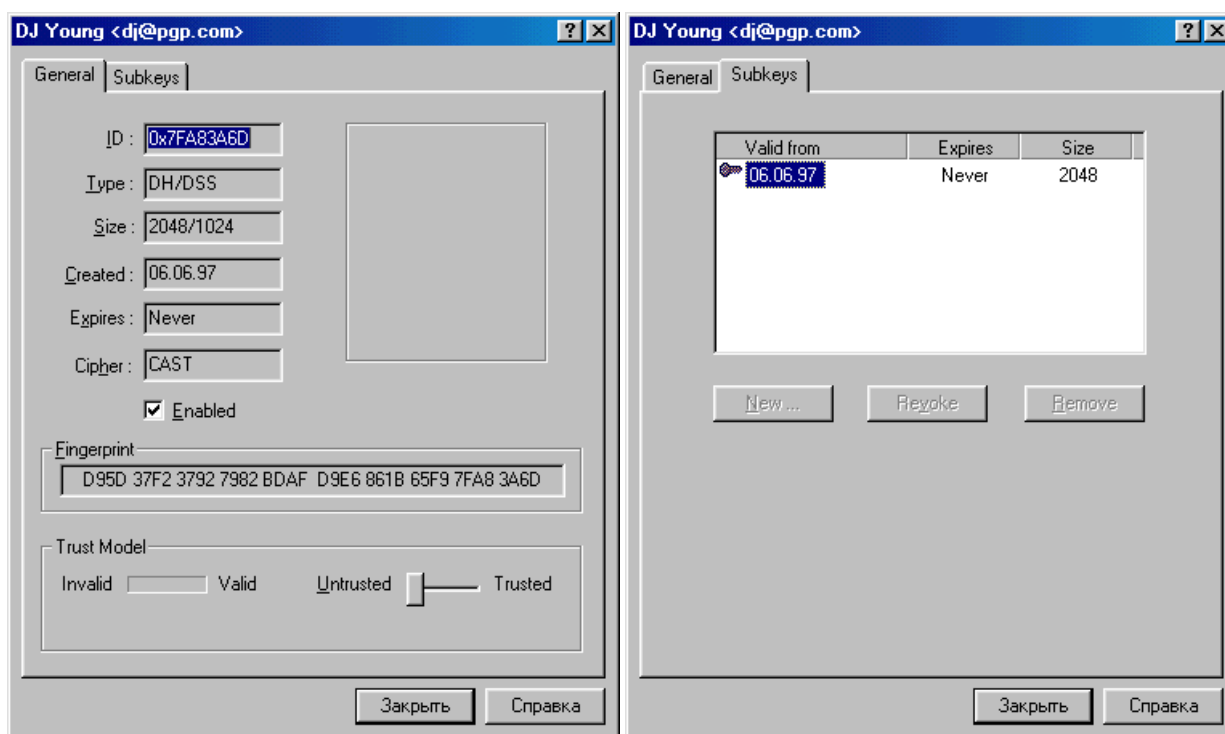
New Key - add new key.



Rice. 8.4. Window Key Generation Wizard

Share Split - divide key between several owners.

Import/Export - import/export key from/to text file. **Key Properties** – properties (in this case you can ask degree trust, key validity), Fig. 8.4, 8.5.



Rice. 8.5. Properties key

The Server menu contains commands for working with a remote key server.

Team Send to allows send keys on selectable server, Fig.8.6.

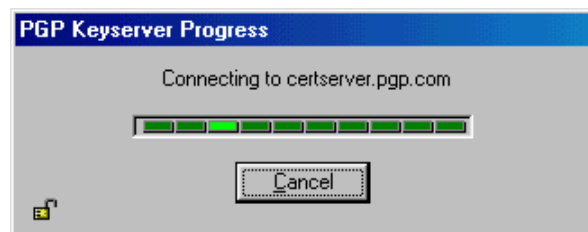
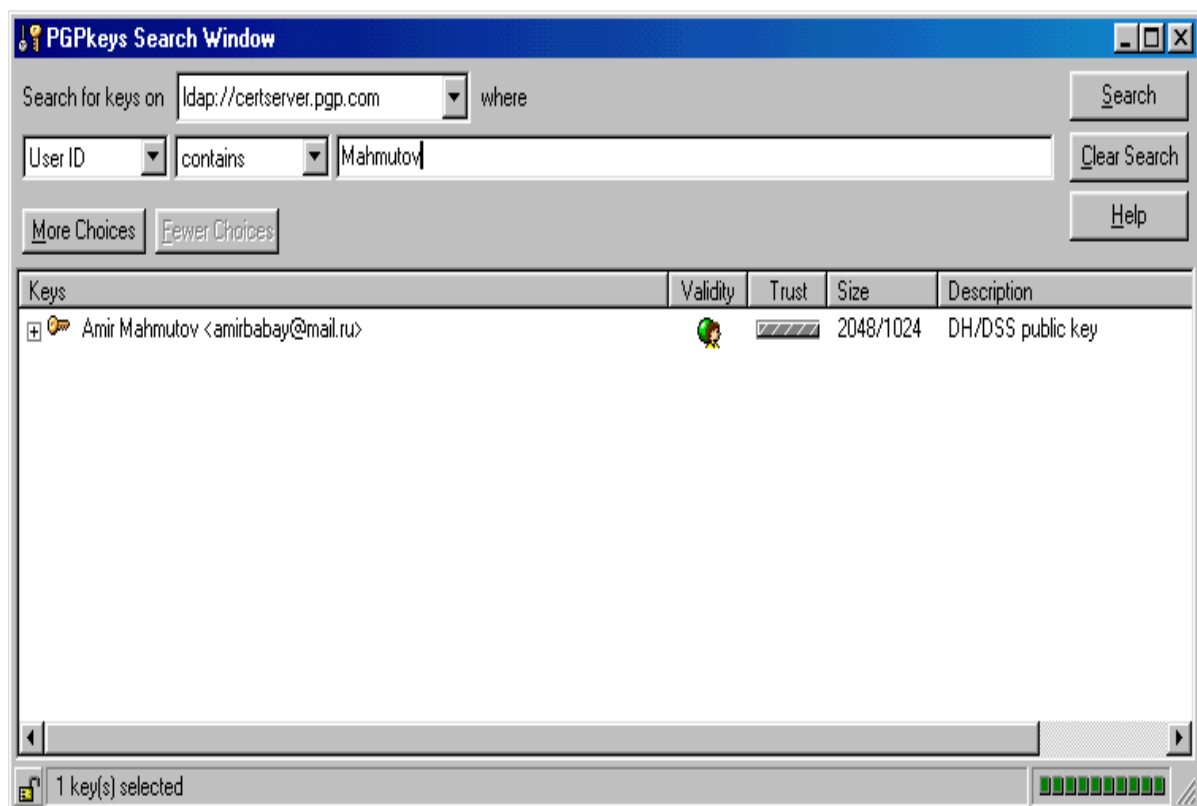


Fig.8.6. Compound With remote server

Using the Search command, you can try to find a key on a remote server by setting the appropriate parameters, Fig. 8.7.



Rice. 8.7. Search key on remote server

Update command allows you to update the selected key by receiving information from the server.

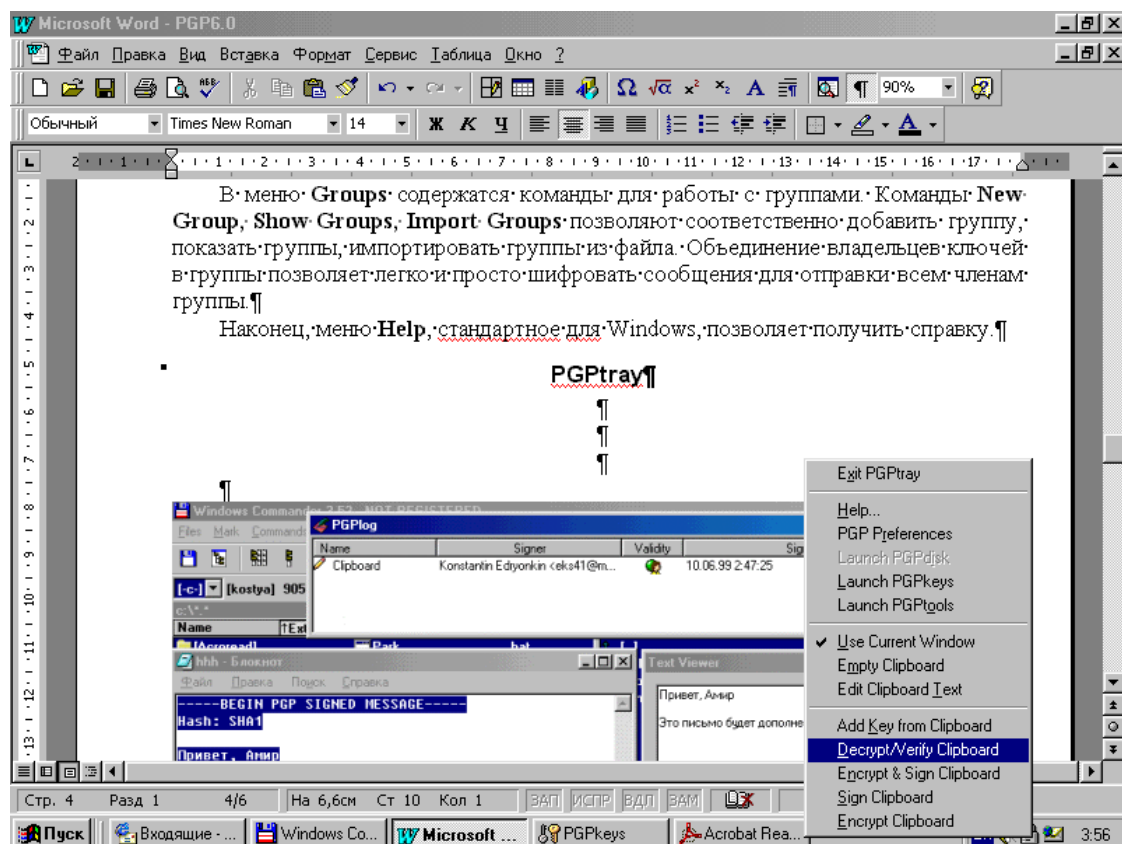
IN menu **Groups** contained teams For work With in groups. Teams **New Group, Show Groups, Import Groups** allow ,

Accordingly, you can add a group, show groups, or import groups from a file. Grouping key owners allows you to easily encrypt messages sent to all group members.

Finally, menu **Help** , standard For *Windows* , allows get the right ku.

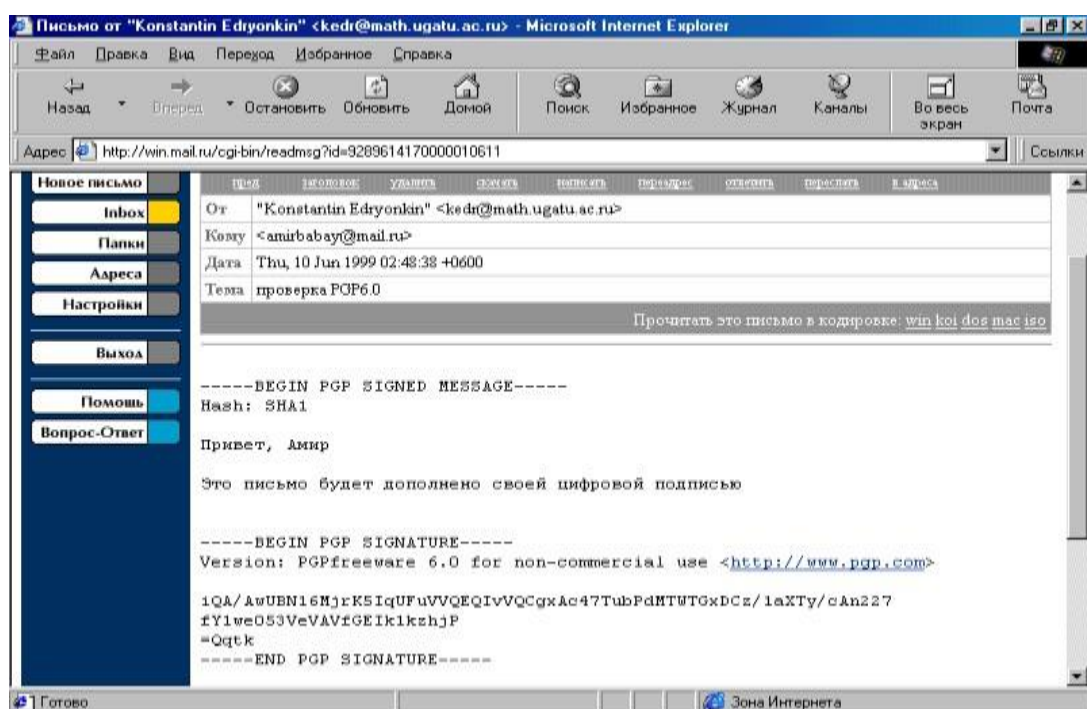
PGPtray

Loads when *Windows starts* . To activate the menu, simply click the icon next to the clock (Fig. 8.8).

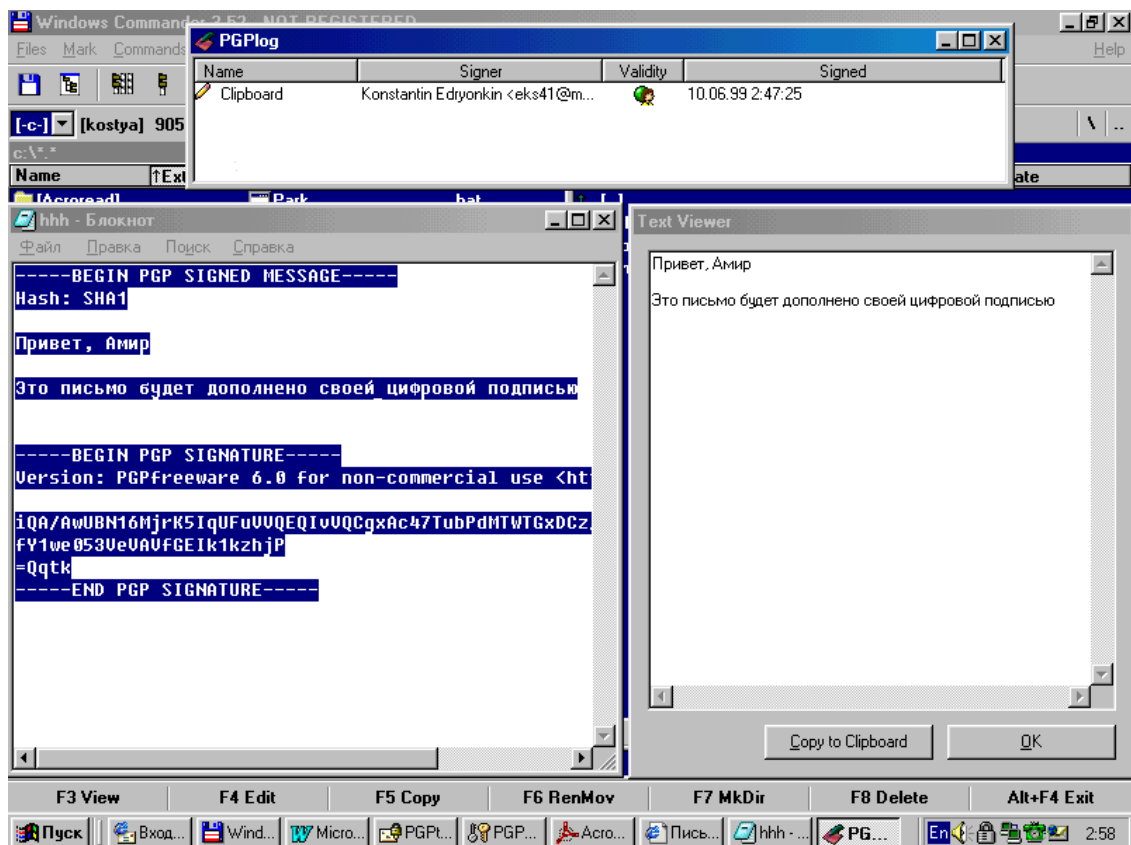


Rice. 8.8. Launch PGP With using icon on panels tasks

The menu contains commands for exiting, launching other *PGP components* , editing the clipboard, working with the clipboard (adding a key, decryption/verification signatures, encryption And signature, Just signature And Just encryption).



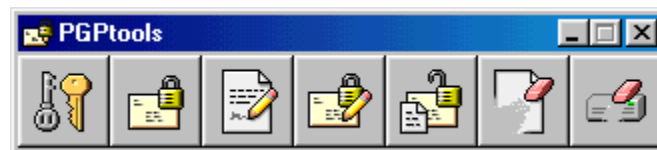
Rice. 8.11. Example receipt letters With digital signature



Rice. 8.12. Results checks signatures

PGPtools

This *PGP component* is a small panel with buttons that allow you to execute necessary action: open *PGPkeys*, encrypt, sign, encrypt and sign simultaneously, decrypt/check signature, rub down file, rub down unused disk space, Fig. 8.13.



Rice. 8.13. Components panels PGPtools

An alternative to *PGPtools* can be the commands added during *PGP* installation to the context menu for working with files and the **File menu** of *Windows Explorer*.

Exercise

1. Get acquainted with information O program PGP (cm. paragraph 1).
2. Launch the PGPtools program (using the Start menu or the PGPtray icon on the taskbar), familiarize yourself with and reflect in the lab report *the composition of the software tools included in the PGP system* (if necessary, use the help about the PGP system).
3. Create cryptographic keys using PGPkeys. Include in your lab report *information on the procedure for creating encryption keys in the PGP system*, and include copies of the screen forms used in this process, as well as answers to the following questions, in the electronic version of the report:
 - **How is the randomness of the selected cryptographic keys ensured in the PGP system ?**
 - **How And Where stored secret key user V system PGP ;**
 - **How can the PGP system ensure the ability to restore a user's secret key if it is accidentally lost ?**
4. Study (on example ordinary text files And files image-

methods of encrypting and decrypting files using the Encrypt and Decrypt functions of the PGPtools program. Include *information on the procedure in the lab report encryption and decryption of files in the PGP system*, include in the electronic version of the report copies of the screen forms used and answers to the questions:

- **which additional parameters encryption can be used, what is their meaning and possible application (required)**
- **check with an example and the results of the inspection should be reflected in the report);**
- **how it is generated, how and where it is stored symmetric file encryption key in the PGP system;**
- **How can access to an encrypted file be ensured by other users ?**
- **whether and how the file size changes after it is encrypted (give specific examples for different types of files) .**

5. Learn (using documents from your folder as an example) how to obtain and verify an electronic digital signature for files using functions Sign and Verify of PGPtools programs. Include in the report *information on the procedure for ensuring the authenticity and integrity of electronic documents in the PGP system*, enable in electronic version report copies of the screen forms used and answers to questions:

- **what additional parameters for obtaining an electronic digital signature can be used, what is their meaning and possible application (be sure to check using an example and reflect the results of the check in the report) ;**
- **What is the program's reaction to a violation of the integrity of a signed document (be sure to check it using an example and reflect the results of the check in the report) .**

6. Study methods simultaneous encryption (decryption) And By-

obtaining (verifying) an electronic digital signature in the PGP system using the Encrypt Sign and Decrypt/Verify functions of the PGPtools program. Include in the report *information on the procedure for simultaneously ensuring the confidentiality, authenticity, and integrity of electronic documents in this system*, and also include in electronic version report copies of the screen forms used in this process .

7. Learn how to securely delete files containing confidential information using the Wipe function of PGPtools. Enable

report *information on the procedure for destroying confidential electronic documents in the PGP system* and include in the electronic version of the report copies of the screen forms used for this purpose.

8. Explore methods for securely destroying residual information that may contain confidential data using the Freespace Wipe feature of PGPtools. Include in the report *information on the purpose and procedure for using this feature*, copies of the screen forms used in it, and answers to the following questions:

- **How is secure destruction of residual confidential information achieved in the PGP system ?**

- **Is such a method of destruction absolutely reliable and, if not, how can absolutely reliable destruction of residual information be ensured?**

9. Learn how to quickly perform PGP system functions using the PGPTray program, the shortcut to which is located on the right side of the taskbar. Include *information about the purpose and procedure for using this program in the report*, and also include copies of the screen forms used in the electronic version of the report.

10. Learn how to manage PGP system settings when used in organizations using the PGAdmin program (go through all the steps of the dialog With master right up to to last one, on which instead of buttons Save

(click the "Cancel" button). Include in the report *information about the capabilities and procedure for administering the PGP system* , include in the electronic version of the report copies of the screen forms used and answers to the questions:

- **What functions for managing encryption and ensuring the integrity of information resources does the PGPAdmin program provide to the administrator ?**

- **What functions for managing PGP users' cryptographic keys does the PGPadmin program provide to the administrator ?**

- **What capabilities does the PGPadmin program provide for managing the PGP program functions available to users and where is such information stored .**

11. Include in the laboratory report the answers to the control questions selected in accordance with the option number specified by the teacher .

Option number	Control questions
1,5,7	How choose length cryptographic key V PGP system ?
2,4,6,8	IN how advantages And flaws cryptographic methods protection information?
11,13,15,17.19	Which computer systems are called safe?
12,14,16	IN how are concluded main requirements To security computer systems?
20,22,24,30	To meet what security requirements for computer systems can cryptographic protection methods be used?
21,23,25	In your opinion, how reliable are the cryptographic information protection methods used in the PGP program?
3,9,18,28	What kind main functions protection information possesses program PGP?

10,26,27, 29	What is the principle behind the secure destruction of residual confidential information feature of PGP?
-----------------	--

List literature

1. Agranovskiy A.V., Hadi R. A. Practical Cryptography. Algorithms and their programming (+ CD-ROM). St. Petersburg, SOLON-Press, 2002.
2. Welschenbach M. Cryptography on Si And C++ V action (+CD-ROM). – M.: Triumph, 2004.
3. Bashly P.N., Babash A.V., Baranova E.K. Information security: a training and practical manual. – M.: Publishing Center of the EAOI, 2010.
4. Coutinho WITH. Introduction V theory numbers. Algorithm RSA. – M.: Postmarket, 2001.
5. Mao IN. Modern cryptography: theory And practice. - M.: Publisher-sky house Williams, 2005.
6. Moldovyan A. A., Moldovyan N. A., Sovetov B. Ya. Cryptography. Textbook for universities – M.: Lan, 2005.
7. Fomichev V.M. Discrete mathematics and cryptology. Well lectures. – M.: Dialog-MIFI, 2003.

LABORATORY JOB No. 9

CORRECTIVE CODES. CODES HEMMING

Objective of the work: To familiarize oneself with the general principles of constructing and using correction codes to control the integrity of information distributed over telecommunication channels.

Note: To complete the lab work, you must install the file on your computer. *Hemming.exe*, which located V archive *Code Hamming.rar*.

1. THEORETICAL INTELLIGENCE

1.1. Methods control correctness transmissions data

1.1.1. Code With verification on parity

Integrity control of information during transmission from source to receiver can be carried out using error-correcting codes.

The simplest corrective code - *code With verification on parity*, which is formed by adding one redundant bit to a group of information bits, the value of which is selected in such a way that the sum of the units in the code combination (that is, the weight of the code combination) is always even.

Example. Let's consider code With verification on parity, formed by adding a check digit to a simple binary code.

	<i>Information</i>	<i>Control</i>
0	000	0
1	001	1
2	010	1
3	011	0
4	100	1
5	101	0
6	110	0
7	111	1

This code detects all single errors and group errors of odd multiplicity, since the parity of the number of units will also be violated in this case.

It should be noted that when coding, it is advisable to make the number of units in the code combination odd and to carry out oddness control; in this case, any combination, including one representing zero, will have at least one unit, which makes it possible to distinguish the complete absence of information from the transmission of zero.

1.1.2. Codes Hamming

N - significant code Hamming has m - information digits and k - control. The number of control digits must satisfy the relation:

$$k \geq \log_2(n + 1),$$

where

$$m \leq n - \log_2(n + 1).$$

The Hamming code is constructed in such a way that the number of control bits calculated according to the above formula, which are formed by, are added to the existing information bits of the code combination. counting parity amounts units For certain groups information bits. When receiving such a code combination from the received information bits And control discharges by way similar counting parity make up a correction number, which is equal to zero if there is no error, or indicates the number of the erroneous digit.

Let's consider more details process coding.

Let the first check digit have an odd serial number. We will set it during encoding in such a way that the sum of the units of all digits with odd serial numbers is equal to 0. Such an operation can be written as a ratio:

$$E_1 = a_1 \oplus a_3 \oplus a_5 \oplus a_7 \dots = 0,$$

where a_1, a_3, a_5 , and a_7 are binary symbols located in digits with numbers 1, 3, 5, 7,...

The appearance of a one in the second digit (from the right) of the correction number indicates an error in those digits of the code combination whose ordinal numbers (2, 3, 6, 7, ...) in the binary image have a one in the second digit from the right. Therefore, the second coding operation, which allows one to find the second check digit, has the form:

$$E_2 = a_2 \oplus a_3 \oplus a_6 \oplus a_7 \dots = 0,$$

Reasoning similar way:

$$E_3 = a_4 \oplus a_5 \oplus a_6 \oplus a_7 \dots = 0, E_4 = a_8 \oplus a_9 \oplus a_{10} \oplus a_{11} \dots = 0,$$

After receiving the code combination, together with the generated check digits, the same counting operations are performed as described above, and the resulting number:

$$E_k E_{k-1} \dots E_2 E_1$$

is considered corrective, and when $E_k E_{k-1} \dots E_2 E_1 = 0$ there are no errors, and if there are errors, those sums E_i in the formation of which the erroneous bit participated are not *equal to zero*.

The correction number in this case will be equal to the ordinal number of this digit.

The location of the check bits in each code combination is determined so that the check bits are used only in one parity calculation. This simplifies the coding process. These positions are integer powers of two: 1, 2, 4, 8, 16, etc.

Example. Let's make it up six-digit code Hamming

$$n = 6, k \geq \log_2 6, k = 3, \quad m = n - k = 3$$

fra	Qi	Simple binary code	Code Hamming				
			6 2	5 1	4	3	
0		000	0	0	0	0	0
1		001	0	0	0	1	1
2		010	0	1	1	0	0
3		011	0	1	1	1	1
4		100	1	0	1	0	1
5		101	1	0	1	1	0
6		110	1	1	0	0	1
7		111	1	1	0	1	0

Options bug fixes

Accepted code: 11 **1** 100 corrected 110100 - error in the correction number in rank 4;

1 **1** 1010 corrected 101010 - error By corrective number V discharge 5;

1 00000 corrected 000000 - error By corrective number V discharge 6

1.1.3. Cyclic codes

Cyclic codes are a type of systematic codes and therefore have all their properties. The characteristic feature of a cyclic code, which determines its name, is that if an n -digit code combination $a_0 a_1 a_2 \dots a_{n-1} a_n$ belongs given code, That And combination $a_n a_0 a_1 a_2 \dots a_{n-1}$, received cyclical rearrangement signs, Also belongs this code.

The idea of constructing cyclic codes is based on the use of *irreducible polynomials*. An irreducible polynomial is one that cannot be represented as a product of lower-degree polynomials, that is, a polynomial that is divisible only by itself or by 1.

When constructing cyclic codes, irreducible polynomials play the role of so-called generator polynomials, the type of which determines the main characteristics of the resulting code: redundancy and error-correcting ability. Table 1.1 lists irreducible polynomials with degrees $k = 1, 2, 3, 4$.

Table 1.1

$P(x)$	$P(1.0)$
$k=1$ $x+1$	11
$k=2$ x^2+x+1	111
$k=3$ x^3+x+1	1011
x^3+x^2+1	1101
$k=4$ x^4+x^3+1	10011
x^4+x^2+1	11001
$x^4+x^3+x^2+x+1$	11111

The basic principles of coding in a cyclic code are as follows. A binary-encoded n -bit number is represented by a polynomial of the $(n - 1)$ th power of some variable x , where the coefficients of the polynomial are the binary signs of the corresponding digits. Writing, reading, and transmitting code combinations in a cyclic code are performed starting with the most significant digit. In accordance with this rule, in what follows, the numbers themselves and the polynomials corresponding to them will be written so that the most significant digit is on the right.

0 1 2 3 4 5

Example: The number 110101 (the digits are numbered from left to right, according to the rule given above, from 0 to 5) will be represented by a fifth-degree polynomial: $1 + x + x^3 + x^5$.

It should be noted that a cyclic permutation of digits in the binary representation of a number corresponds to the multiplication of a polynomial by x , in which x^n is replaced by 1 and goes to the beginning of the polynomial.

Example: Let's multiply the polynomial obtained in the previous example by x . We transform the new polynomial $x + x^2 + x^4 + x^6$ by replacing x^6 with 1. We finally get $1 + x + x^2 + x^4$, which corresponds to the number 111010.

The cyclic code of an n -digit number, like any systematic code, consists of m information and k check digits, with the latter occupying the k least significant digits. Since the sequential transmission of code combinations is carried out, as we have already indicated, starting with the most significant digits, the check digits are transmitted at the end of the code.

As already indicated, the code is formed using the so-called generating polynomial, $P(x)$, of degree k , the type of which determines the basic properties of the code – redundancy and corrective ability.

A code polynomial, $F(x)$, is a polynomial of degree less than $(m + k)$ if it is divisible by the generator polynomial $P(x)$ without remainder. After the message has been transmitted, decoding consists of dividing the polynomial $H(x)$, corresponding to the received code, by $P(x)$. In the absence of errors, $H(x)=F(x)$, and the division is being carried out without remainder. Availability non-zero remainder indicates on That, that information was distorted during transmission or storage.

To obtain a systematic cyclic code, the following relationship is used:

$$F(x) = x^k G(x) \square R(x),$$

where $G(x)$ - polynomial, representing informational symbols
(informational polynomial);

$R(x)$ - remainder from division $x^k G(x)$ on $P(x)$.

Example. Let's consider coding 8-digit numbers 10110111. Let the generator polynomial of the 3rd degree $P(x) = 1 + x + x^3$ be given for coding.

Делим $x^3 G(x)$ на $P(x)$: $G(x) = 1 + x^2 + x^3 + x^5 + x^6 + x^7$;
 $x^3 G(x) = x^3 + x^5 + x^6 + x^8 + x^9 + x^{10}$;

$$\begin{array}{r} x^{10} + x^9 + x^8 + x^6 + x^5 + x^3 x^{10} + x^8 + x^7 \\ \underline{x^9 + x^7 + x^6 + x^5 + x^3 x^9 + x^7 + x^6} \\ x^5 + x^3 \\ \underline{x^5 + x^3 + x^2 R(x) = x^2} \end{array} \quad \begin{array}{r} | 1 + x + x^3 \\ \hline x^7 + x^6 + x^2 \end{array}$$

Using ratio to obtain systematic cyclic code, we find $F(x)$:

$$F(x) = (x^3 + x^5 + x^6 + x^8 + x^9 + x^{10}) \square x^2.$$

So, the final code combination is, corresponding to $F(x)$, has the form

$$\begin{array}{r} 00010110111 \\ 001 \\ \hline \end{array}$$

контрольные разряды $\rightarrow$ 001 10110111

In practice, the coding procedure employed is even simpler. Since we are interested only in the remainder and the quotient is not used in the final result, we can sequentially subtract the divisor from the dividend and the resulting differences modulo 2 until the difference has a lower power than the divisor. This difference is the remainder. Such an algorithm can be implemented in hardware using a k-bit shift register with feedback. Obviously, the cyclic code obtained in this way will be systematic.

However, there is a second option for obtaining a cyclic code, when the next code combination is obtained by multiplying the code combination $C(x)$ of a simple n-digit code by the generating polynomial $P(x)$.

In the second method of forming cyclic codes, the information and control symbols in the cyclic code combinations are not separated from each other, which complicates the process decoding. Therefore This encoding method is used less frequently than the first. Let's consider an example of encoding using the second option, where we will directly use the binary recordings of the original code combinations when performing operations.

Example. Given a generator polynomial of the form $P(1,0)=1101$. It is required to construct a cyclic code from a simple four-digit code using the second method. As an example for construction, we will use the initial combination $C(1,0)=0011$. Operation multiplication this combinations on the generating polynomial will be written as follows:

$$\begin{array}{r}
 0011 \\
 1101 \\
 \hline
 0011 \\
 \oplus 0011 \\
 0011 \\
 \hline
 0010111
 \end{array}$$

— это и есть циклический код для 0011.

The remaining code combinations of the non-systematic cyclic code, given in Table 1.2, were obtained in a similar manner.

Table 1.2

<i>Simple character code</i> $C(x)$	<i>four-</i>	<i>Cyclic (7,4) – code</i> $C(x)$ $P(x)$, where $P(1,0)=1101$
0000		0000000
0001		0001101
0010		0011010
0011	*	0010111 *
0100		0110100
0101		0111001
0110		0101110
0111		0100011
1000		1101000
1001		1100101
1010		1110010
1011		1111111
1100		1011100
1101		1010001
1110		1000110
1111		1001011

The problem of detecting various types of errors using cyclic code, as already indicated, comes down to finding the necessary generator polynomial. It is not our purpose to consider this issue; the basic principles of selecting generator polynomials are described in sufficient detail in the textbook .

Effective coding information

1.1.4. General provisions

Let us remember that *encoding* is the presentation of messages in a form convenient for transmission over a given channel, and *decoding* is the restoration of information By accepted signal. One from most important questions coding- nia is increase his efficiency, That There is by way elimination redundant

The average number of characters required per letter in a message is reduced. This type of coding is called *efficient coding*. It follows from the above that efficient coding solves the problem of maximum *information compression*.

Taking into account the statistical properties of the information source, it is possible to minimize the average number of binary symbols required to express one letter of the message, which, in the absence of interference, makes it possible to reduce the message transmission time and its volume.

Efficient coding is based on *Shannon's fundamental theorem for an interference-free channel*⁴, the essence of which is as follows.

³ Werner M. Basics coding. Textbook For Universities. – M.: Technosphere, 2006. - 288 With.

⁴ Shannon, K. Mathematical Theory of Communications. Works on Information Theory and Cybernetics: translated from English; edited by R.L.Dobrushin and O.B.Lupanov. Moscow: IL, 1963. 512 p.

The essence theorems Shannon

Сообщения, составленные из букв некоторого алфавита, можно закодировать так, что среднее число двоичных символов на букву сколь угодно близко к энтропии источника этих сообщений, но не меньше этой величины.

The presence of redundancy in messages raises the question of data compression, that is, transmitting the same amount of information using shorter character sequences— *lossless compression methods*. For this purpose, special compression algorithms are used to reduce redundancy. The compression effect is assessed by *the compression ratio*.

$$K = n/q,$$

Where n - number minimally necessary symbols For transmissions message-
nia;

q - number symbols V message, compressed data method.

So, at effective binary coding n equals entropy source
information.

Along with compression methods that do not reduce the amount of information in a message, compression methods based on the loss of insignificant information—*lossy compression methods*—are used. Note that lossy compression methods are unacceptable for certain types of information, such as text or numeric data.



Rice. 1.1. Methods compression information

The simplest way to compress numerical information represented in *ASCII code* is to use a shortened code with four bits per character instead of eight, since in *ASCII* encoding, digits, as well as the symbols "*period*" , "*comma*" and "*space*" have the first code character a zero that can be discarded.

is the RLE (Run Length Encoding) algorithm due to its simplicity . where instead of transmitting a string of identical characters, a character and the length of the string are transmitted. This method is effective when transmitting raster images, especially monochrome ones, where there are a large number of strings of 1s and 0s, but is of little use when transmitting text. The *RLE algorithm* is implemented in the *PCX format*.

Пример использования алгоритма RLE

Исходный текст (9 байт): 77 77 77 77 11 11 11 01 FF

В исходной последовательности:

1. Выделяется повторяющаяся последовательность байт
2. Заменяется счетчиком повторов (04) и кодирующим байтом (77)
3. 00 – нет повторений
4. 02 - сколько байт не повторяется
5. Какие байты не повторяются (01 FF)

Результат сжатия (8 байт): 04 77 03 11 00 02 01 FF

Недостатками алгоритма *RLE* являются:

- низкая степень сжатия;
- сильная зависимость от количества повторов байт в исходной информационной последовательности.

Достоинство – простота реализации.

Main sphere use methods compression With losses - graphic, audio and video files, where possible, partially sacrificing quality characteristics information, achieve significant reduction of its volumes.

The most widely used lossless data compression methods are statistical compression algorithms that take into account the probability of occurrence of individual symbols in the information stream. These are primarily the Shannon-Fano and Huffman algorithms.

1.1.5. Algorithm Shannon- Fano

Efficient coding using the Shannon-Fano method is based based on Shannon's fundamental theorem for an interference-free channel.

Algorithm effective coding By Shannon- Fano

- The letters of the message alphabet are written out in the table in order of decreasing probability.

- They are then divided into two groups so that the sums of the probabilities in each group are as equal as possible.
- All letters in the upper half are assigned 1 as the first symbol, and all letters in the lower half are assigned 0.
- Each of the resulting groups is in turn divided into two subgroups with the same total probabilities, etc. The process is repeated until each subgroup contains one letter.

Example . Let us consider an alphabet of 7 letters, assigning to each letter a probability of its appearance in a message p_i (there are special tables of the probability of the appearance of letters of the Russian or Latin alphabets in messages, for example, see Appendix 1.2.1).

Буква	Вероятность, p_i	Процесс получения кода	Код Шеннона
A	1/4	1 $\xrightarrow{\text{II}}$	11
E	1/4	1 $\xrightarrow{\text{I}}$ 0	10
F	1/8	0 1 $\xrightarrow{\text{III}}$	011
C	1/8	0 1 $\xrightarrow{\text{II}}$ 0	010
B	1/8	0 0 $\xrightarrow{\text{III}}$ 1	001
D	1/16	0 0 0 $\xrightarrow{\text{IV}}$ 1	0001
G	1/16	0 0 0 0	0000

$$H(\xi) = \sum_{k=1}^N p_k \log_2 1/p_k = 2/4 + 2/4 + 3/8 + 3/8 + 3/8 + 4/16 + 4/16 = 2,625.$$

It is clear that with conventional binary coding, without taking statistical characteristics into account, three symbols are required to represent each letter. The greatest compression effect is achieved when the letter probabilities are integer negative powers of two. The average number of symbols per letter in this case is exactly equal to the entropy.

Average number symbols on letter

$$L = \sum_{i=1}^N p_i n_i,$$

Where p_i - probability appearances i - th symbol alphabet;

n_i - quantity symbols V code combinations of i - th symbol alphabet. For the example considered in the table

$$L = 1/4 * 2 + 1/4 * 2 + 1/8 * 3 + 1/8 * 3 + 1/8 * 3 + 1/16 * 4 + 1/16 * 4 = 2,625.$$

Difference quantities $(L - H)$ - There is *redundancy code*, and the value $(L - H)/L$ - *relative redundancy*.

Redundancy can be interpreted as a measure of the amount of wasted alternative choices. Since, in practical cases, individual characters never occur equally frequently, constant-length encoding is redundant in most cases. Despite this, such encoding is used quite often for technical reasons.

1.1.6. Algorithm Huffman

Algorithm effective coding By Huffman

The letters of the message alphabet are written out in the main column of the table in descending order of probability.

- The last two letters are combined into one auxiliary letter, to which the total probability is assigned.
- The probabilities of the letters that were not involved in the combination and the resulting total probability are again arranged in descending order of probabilities, and the last two are combined, and so on, until they get a single auxiliary letter with probability 1.
- Next For constructions code is used binary tree, V root ko-

which is located a letter with probability 1, when the branch branches, the code 1 is assigned with a higher probability, and the code 0 with a lower probability (possibly the left one is 1, and the right one is 0).

Example: Let us consider a conditional alphabet of eight letters, each of which is assigned a corresponding probability of its appearance in a message.

Буквы	Вероятности	Вспомогательные столбцы вероятностей	Код Хаффмена
Z1	0,22	0,22 0,22 0,26 0,32 0,42 0,58 } 1	01
Z2	0,20	0,20 0,20 0,22 0,26 0,32 } 0,42 }	00
Z3	0,16	0,16 0,16 0,20 0,22 } 0,26 }	110
Z4	0,16	0,16 0,16 0,16 } 0,20 }	111
Z5	0,10	0,10 0,16 } 0,16 }	100
Z6	0,10	0,10 } 0,10 }	1011
Z7	0,04 }	0,06 }	10101
Z8	0,02 }		10100

$$L = 0,22 \times 2 + 0,20 \times 2 + 0,16 \times 3 + 0,16 \times 3 + 0,10 \times 3 + 0,10 \times 4 + 0,04 \times 5 + 0,02 \times 5 = 2,8;$$

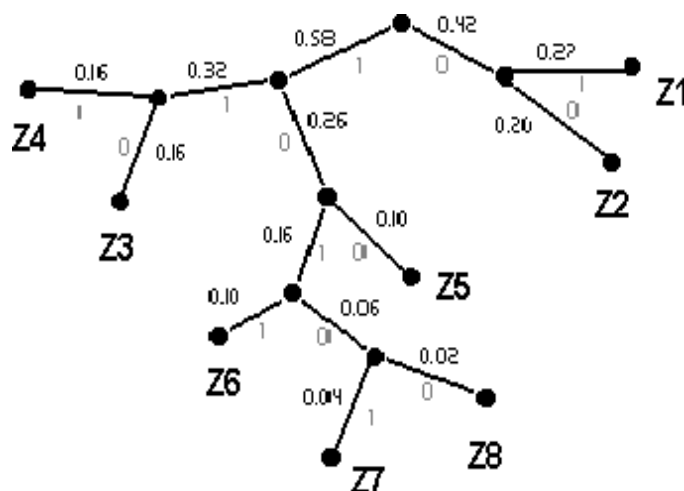
$$H = 2,76;$$

$$L - H = 0,04.$$

For

comparison: in the Shannon code with the same probability distribution $L - H = 0.08$.

Construction binary tree



Appendix 1.2.2 contains tasks for independent work on coding information using the Shannon-Fano and Huffman algorithms .

Appendix 1.2.1

Probabilities separate letters *V* Russian language

Letter	Probability	Letter	Probability
_	0.175	I	0.018
ABOUT	0.090	Y	0.016
E	0.072	Z	0.016
A	0.062	Ъ,ъ	0.014
And	0.062	B	0.014
T	0.053	G	0.013
N	0.053	Ch	0.012
S	0.045	Y	0.010
R	0.040	X	0.009
In	0.038	F	0.007
L	0.035	Yu	0.006
K	0.028	W	0.006
M	0.026	C	0.004
D	0.025	SCH	0.003
P	0.023	E	0.003
U	0.021	F	0.002

Appendix 1.2.2

Tasks For independent work

1. Write a procedure that provides efficient coding: option A) code Shannon; option b) code Huffman, And count average the amount of information per character H ; redundancy $(L - H)$; and the relative redundancy of the resulting code $(L - H) / L$.

For testing programs use two control options:

Option 1

Letters	Probabilities	Shannon code
Z1	0.22	11
Z2	0.20	101
Z3	0.16	100
Z4	0.16	01
Z5	0.10	001
Z6	0.10	0001
Z7	0.04	00001
Z8	0.02	00000 $H = 2.76$ $L = 2.84$ $L - H = 0.08$

Option 2

Letters	Probabilities	Shannon Code
Z1	1/2	1
Z2	1/4	01
Z3	1/8	001
Z4	1/16	0001
Z5	1/32	00001
Z6	1/64	000001
Z7	1/128	0000001
Z8	1/128	0000000 = 1 63/64 L - H = 0

0.1. 2. Source messages generates signs *A*, *IN*, *WITH* With probabilities 0.7; 0.2;

Write procedure allowing:

- define entropy source;
- indicate For these three signs optimal binary coding:

option a) Shannon code; option b)

code Huffman,

And define average length code combinations;

- encode All couples *AA*, *AB*, ...;
- build For these nine steam optimal binary code:

option a) Shannon code; option b)

code Huffman.

• increase blockiness code to three-character combinations, And construct an optimal binary code:

option a) Shannon code;

option b) code Huffman.

Do conclusion about change redundancy code With increase blockiness. How does this affect code efficiency?

Letter	Probability	Code	Sign	Probability	Pair code
A	0,7	0	AA	0,49	1
B	0,2	11	AB	0,14	011
C	0,1	10	BA	0,14	010
			AC	0,07	0011
			CA	0,07	0010
			BB	0,04	0001
			BC	0,02	00001
			CB	0,02	000001
			CC	0,01	000000

$$H = 1,1571;$$

$$L = 1,3;$$

$$L_{\text{sign}} = 2,33;$$

$$L_{\text{symbol}} = L_{\text{sign}} / 2 = 1,165.$$

Conclusion : When coding in large blocks, it is possible to reduce the redundancy of the code; theoretically, the minimum redundancy can be achieved when coding blocks that include an infinite number of letters:

1.2.3. LZW - compression

One of the most widely used lossless compression methods today is the Lempel-Ziv-Welch algorithm .

The compression method was proposed in 1977, an improved (*Terry Welch*) version was published in 1984, and was subsequently modernized several times. This algorithm is used, in particular, by the standard *UNIX Compress procedure* . Algorithms of the group *LZ* (*LZ77*, *LZ78*, *LZSS*) are the basis of almost all modern software and hardware data compression tools. *PKZIP*, *LHA*, *Stacker*, *SuperStor*, *Dblspace* and many other archivers use one or another modification of the *LZ algorithm* .

The compression idea proposed by Lempel and Ziv is as follows: if a sequence of two previously encountered symbols appears in the text of a message, then this sequence is declared a new symbol, and a code is assigned to it, which under certain conditions can be significantly shorter than the original sequence. Subsequently, in the compressed message instead of original sequences is being recorded appointed code.

When decoding, similar actions are repeated and therefore the sequences of symbols for each code become known.

Process coding

1. Each symbol of the original alphabet is assigned a specific code (here code is a serial number, starting from 0).
2. It is selected first symbol messages And is replaced on his code.
3. The next two characters are selected and replaced with their codes. At the same time, this two-character combination is assigned its own code (usually a serial number equal to the number of codes already used). So, if the alphabet includes 8 characters with codes from 000 to 111, then the first two-character combination will receive the code 1000, the next – the code 1001, and so on.
4. The next 2, 3, ... N-symbol combinations are selected from the source text until a previously unseen combination is formed. Then this combinations is assigned another one code, And because totality If one of the first N-1 symbols has already been encountered, then it has its own code, which is written in place of these N-1 symbols. Each act of introducing a new code is called an encoding step.
5. Process continues to exhaustion original text.

Process decoding

When *decoding*, the code of the first symbol, and then the second and third, are replaced by alphabetic characters. In this case, the code for the combination of the second and third symbols becomes known. The next position can only contain codes already known symbols and their combinations. The decoding process continues until the compressed text is exhausted.

Example .

Let the source text be a binary code (the first row of Table 2.3), that is, the source alphabet $A=\{1,0\}$. The codes of these symbols are also 0 and 1, respectively.

The *LZ code* generated by the Lempel-Ziv method is shown in the second row of Table 2.3. The third row indicates the coding steps after which the transition to the representation of *A codes* with an increased number of digits r occurs. Thus, in the first step, the code 10 is introduced for the combination 00 and therefore in the following in two steps $r = 2$, after third step $r = 3$, after seventh step $r = 4$. IN in the general case $r=k$ after step $2^{k-1} - 1$.

In the given example, the *LZ code* turned out to be even longer than the original code, Since short texts with a small number of repeating characters usually do not produce a compression effect, the compression effect is noticeable in fairly long texts and is especially noticeable, for example, in graphic files.

Table 1.5

Original text	0.00.000. 01. 11. 111.1111. 110. 0000.00000. 1101. 1110.
LZ code	0.00.100.001.0011.1011.1101. 1010.00110.10010.10001.10110.
r	2 3 4
Introduced codes	- 10 11 100 101 110 111 1000 1001 1010 1011 1100

An improved compression and decompression algorithm using the Lempel-Ziv-Welch method is shown in Fig. 1.2. Fig. 1.3 shows an example of a software implementation of the *LZW algorithm*.

Алгоритм компрессии

```
СТРОКА = очередной символ из входного потока
WHILE входной поток не пуст DO
    СИМВОЛ = очередной символ из входного потока
    IF СТРОКА + СИМВОЛ в таблице строк THEN
        СТРОКА = СТРОКА + СИМВОЛ
    ELSE
        вывести в выходной поток код для СТРОКА
        добавить в таблицу строк СТРОКА + СИМВОЛ
        СТРОКА = СИМВОЛ
    END of IF
END of WHILE
вывести в выходной поток код для СТРОКА
```

Алгоритм декомпрессии

```
Читать СТАРЫЙ_КОД
вывести СТАРЫЙ_КОД
WHILE входной поток не пуст DO
    читать НОВЫЙ_КОД
    СТРОКА = перевести НОВЫЙ_КОД
    вывести СТРОКУ
    СИМВОЛ = первый символ СТРОКИ
    добавить в таблицу перевода СТАРЫЙ_КОД + СИМВОЛ
    СТАРЫЙ_КОД = НОВЫЙ_КОД
END of WHILE
```

Fig.1.2. Algorithm *LZW* compression And decompression

Введите текст:
///wwwrrrttet

Код
/ #256 W #258 R #260 T T E T

[-Сжать=]

[=Распаковать=]

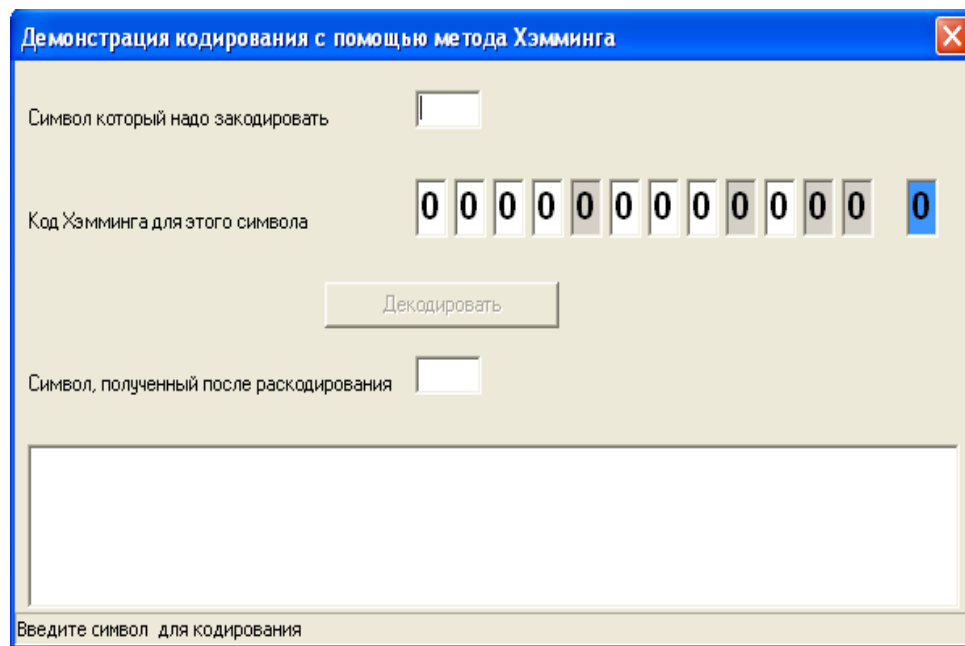
///wwwrrrttet

№	Вход новый код	Старый код	Строка выход	Символ	Новый код таблицы
1	/	/	/		
2	256	/	//	В	//
3	W	256	W	W	//W
4	258	W	WW	!	WW
5	R	258	R	R	WWR
6	260	R	RR	!	RR
7	T	260	T	T	RRT
8	T	T	T	T	TT
9	E	T	E	E	TE
10	T	E	T	T	ET

Fig.1.3. Example software implementation algorithm

1. Description laboratory work

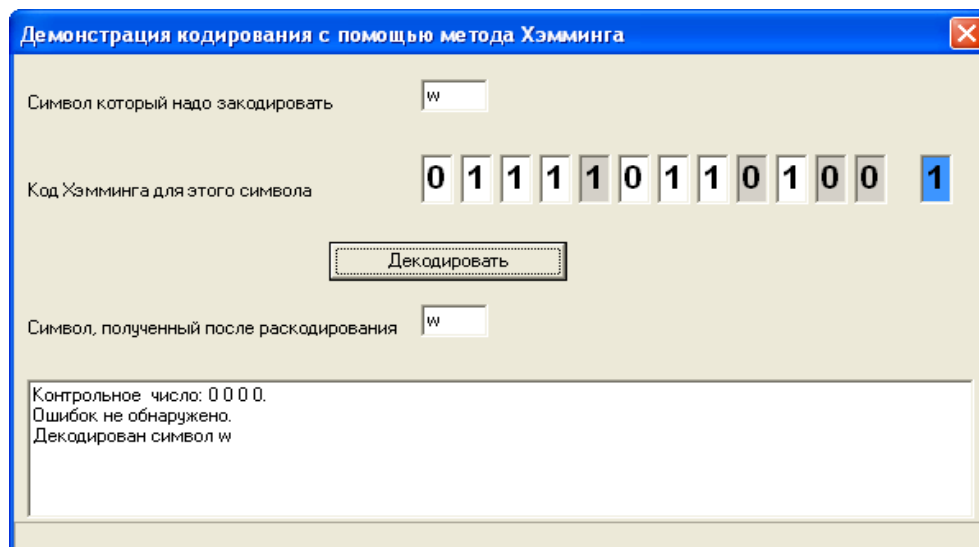
The program is designed for encoding symbols using the Hamming algorithm. The main program window is shown in Fig. 9.1.



Rice. 9. 1. Main window programs

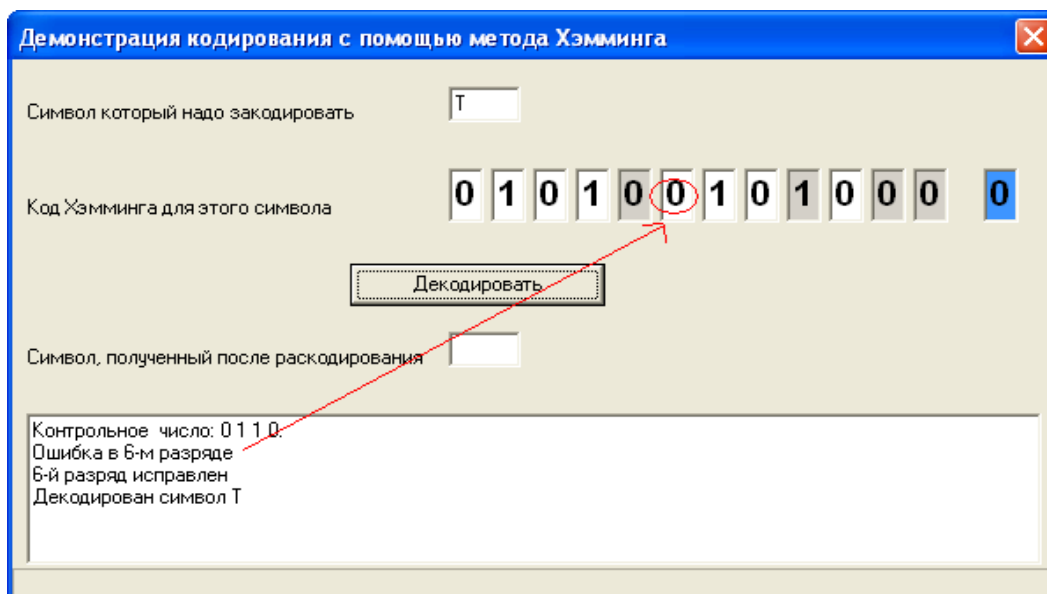
ASCII characters are encoded using a thirteen-character Hamming code containing 8 information bits, 4 check bits and a total parity bit.

Processes coding And decoding presented on Fig. 9.2



Rice. 9. 2. Process coding And decoding

The check bits are marked in gray, and the parity bit is marked in blue. Errors can be introduced into the resulting code. Single errors can be corrected, while double errors are detected without correction. If an error has been corrected, the bit in which it occurred is indicated. The symbol is then decoded. The process of correcting a single error is shown in Figure 9.3.



Rice. 9. 3. Process corrections single mistakes

If an error of double or greater multiplicity occurs, a message is displayed stating that the code cannot be corrected.

2. Exercise

1. Encode the proposed alphabet using the Hamming code (options are given in Table 9.1).

2. In each row of the table with encoded information, enter a single error and record the decoding result in the code table.

3. IN latest two lines tables With coded information introduce double errors, record the decoding result in the code table.

4. Analyze the obtained results and formulate reasoned conclusions.

Describe received code Hamming :

- quantity control And informational discharges And their numbers;
- redundancy code;
- relative redundancy;
- minimum code distance;
- evaluate the correction ability of the obtained code.

5. Make a word from the given alphabet that is at least 5 characters long and encode it using the obtained code Hamming.

Count length source text (*ASCII encoding*) and encoded text (Hamming code).

6. Introduce brief theoretical material, V which describe a method of encoding and decoding information using the Hamming code.

7. Bring results conducted experiments By coding With using the *Hemming.exe program*.

8. Evaluate the results of detection and correction of single and double errors. Draw conclusions about the corrective ability of the research.

blown code.

9. Bring V report answers on control questions, V in accordance with the option number indicated by the teacher.

Options tasks presented V Table 9.1 Table 9.1

Room options comrade	Original alphabet
1,5,9,13,17	Cyrillic A...M
2,6,10,14,18,22	Cyrillic O..Ya
3,7,11,15,19,23,27	Latin A..N
4,8,12,16,20,24,28	Latin O..Z
21,25,29,26,30	Decimal numbers 0..9

Option number	Control questions
1,5,7,3,9,18,28	What represents by yourself And How is used code Hamming?
2,4,6,8,20,22,24,26:30	How is being compiled code Hamming?
11,13,15,10,17,19,27	Which mistakes Can to correct With help code Hamming? From what kind properties code depends his corrective ability ?
12,14,16,21,23,25,29	Which errors may arise when compiling the Hamming code ?

LABORATORY JOB No. 10

CORRECTIVE CODES. CYCLIC CODES

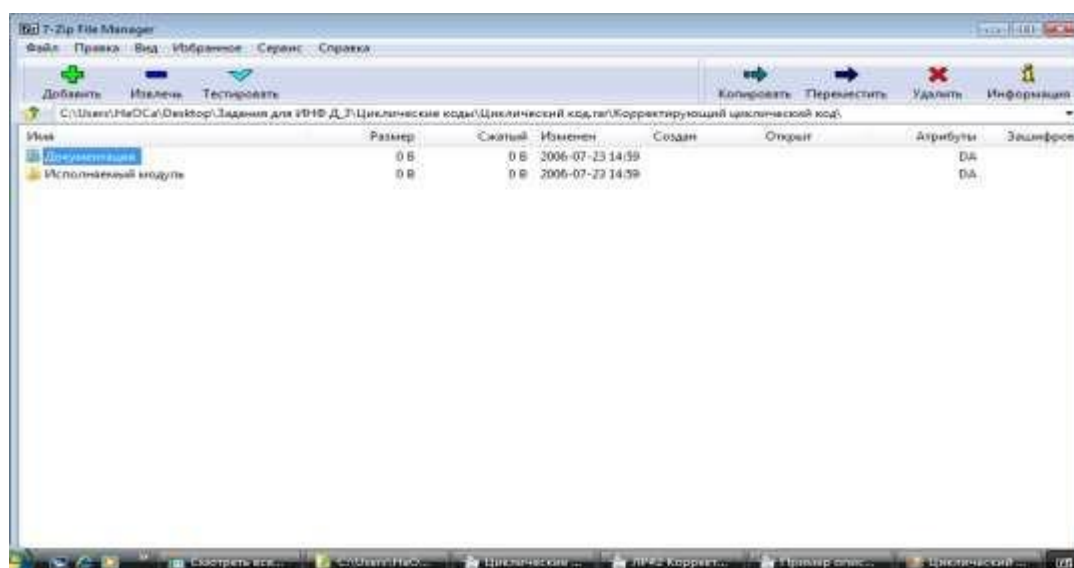
Objective of the work: To become familiar with the general principles of constructing and using error-correcting codes to control the integrity of information distributed over telecommunication channels. To study the principles of constructing cyclic codes.

Note : To complete the laboratory work to work on a computer, you need to install a file

CyclicCode74.exe, which located V archive *Cyclical code.rar*.

1. Description laboratory work

1. Study the information about the program for encoding information using a cyclic code (*Documentation folder*). Run the encoding program (*Executable module folder*, *CyclicCode74.exe* file), Fig. 10.1.



Rice. 10.1. Launch programs *CyclicCode74*

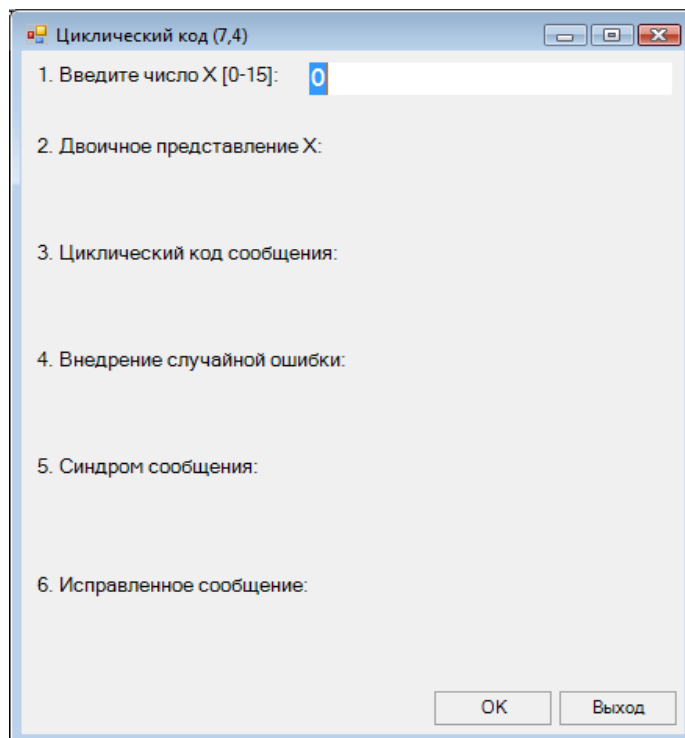
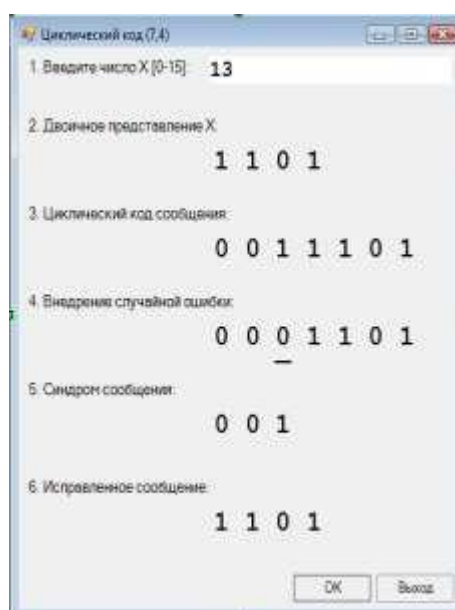


Fig.10. 2. Main program window

2. Encode the proposed option using a cyclic code (*See task options*). To do this, enter the specified number in the line "1. Enter the number X [0-15]" . Save the code table and move it to your personal folder (to do this, press Ctrl+Alt+PrtScSysRq and then Ctrl+V).



Rice. 10. 3. Example filling fields main windows programs

3. Analyze the obtained results and formulate reasoned conclusions.

2. Exercise

1. Obtain a systematic cyclic code using the generator polynomials given in Table 10.1, according to the option specified by the teacher:

Table 10.1

Option	Generating polynomial 3rd degrees
1,7,13,19,25	1011
2,8,14,20,26	1101
3,9,15,21,27	1011
4,10,16,22,28	1101
5,11,17,23,29	1011
6,12,18,24,30	1101

Example: Let's consider the encoding of the eight-digit number 00011111. Let the encoding be given a generator polynomial of degree 3.

$$P(1,0) = 1101, P(x) = 1 + x + x^3$$

Let's divide $x^3 G(x)$ on $P(x)$, Where information polynomial $G(x) = x^3 + x^4 + x^5 + x^6 +$

$$x^7 x^3 G(x) = x^6 + x^7 + x^8 + x^9 + x^{10}$$

$$x^{10} + x^9 + x^8 + x^7 + x^6 \quad x^{10} + x^8 + x^7$$

$$\underline{x^9 + x^6}$$

$$x^9 + x^7 + x^6$$

$$\underline{x^7 + x^5 + x^4}$$

$$x^5 + x^4$$

$$\underline{x^5 + x^3 + x^2}$$

$$x^4 + x^3 + x^2 \quad x^4 + x^2 + x$$

$$\underline{x^3 + x}$$

$$\begin{array}{r} x^3 + x + 1 \\ \hline x^7 + x^6 + x^4 + x^2 + x + 1 \end{array}$$

$$\frac{x^3 + x + 1}{R(x) = 1}$$

We find meaning code polynomial $F(x)$:

$$F(x) = (x^6 + x^7 + x^8 + x^9 + x^{10}) \div 1$$

Like this in this way, finally code combination

$F(x)$ has view :

Control 100 digits

0	1	2	3	4	5	6	7	8	9	10
						1	0	0	0	0
							1	1	1	1

10000011111

10000011111

2. Present in the report a brief theoretical material in which to describe the method of encoding and decoding information using a cyclic code.
 3. Present the results of the experiments conducted on coding using the *CyclicCode74.exe* program .
 4. Evaluate the results of single-error detection and correction. Draw conclusions about the error-correcting capability of the code under study.
 5. Provide answers to the test questions in the report, in accordance with the option number specified by the teacher.
- Options tasks given V Table 10.2.

Table 10.2

<i>Option</i>	<i>Number</i>
1.16	1
2.17	2
3.18	3
4.19	4
5.20	5
6.21	6
7.22	7
8.23	8
9.24	9
10.25	10
11.26	11
12.27	12
13.28	13
14.29	14
15:30	15

Number options	Control questions
1,5,7, 3,9,18,28	Specify the differences between the concepts: <i>information coding</i> , <i>information encryption</i> .
2,4,6,8, 20,22,24, 26:30	Define the concepts: <i>uniform codes</i> , <i>non-uniform codes</i> , <i>prefix coding</i> , <i>separable coding</i> , <i>systematic codes</i> , <i>non-systematic codes</i> .
11,13,15, 10,17,19, 27	From What characteristics of the code does it depend on? its corrective ability?
12,14,16 21,23,25, 29	Compare Hamming codes and cyclic codes in terms of efficiency in information integrity control.

LABORATORY JOB No. 11

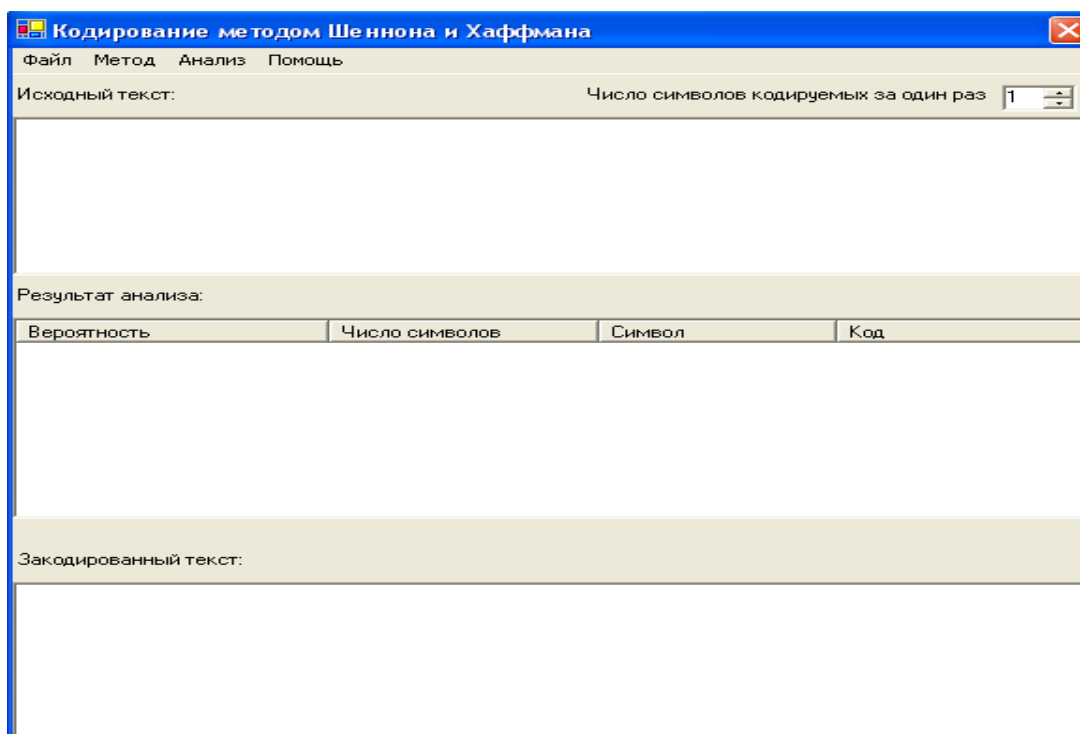
METHODS COMPRESSION BY SHANNON AND HUFFMAN

Objective of the work: To become familiar with the statistical principles of information compression using the Shannon-Fano and Huffman methods.

Note: To complete the lab work, you must install the *Shannon-Huffman.exe* file on your computer, which is located in the *Shannon and Huffman Compression Methods.rar* archive.

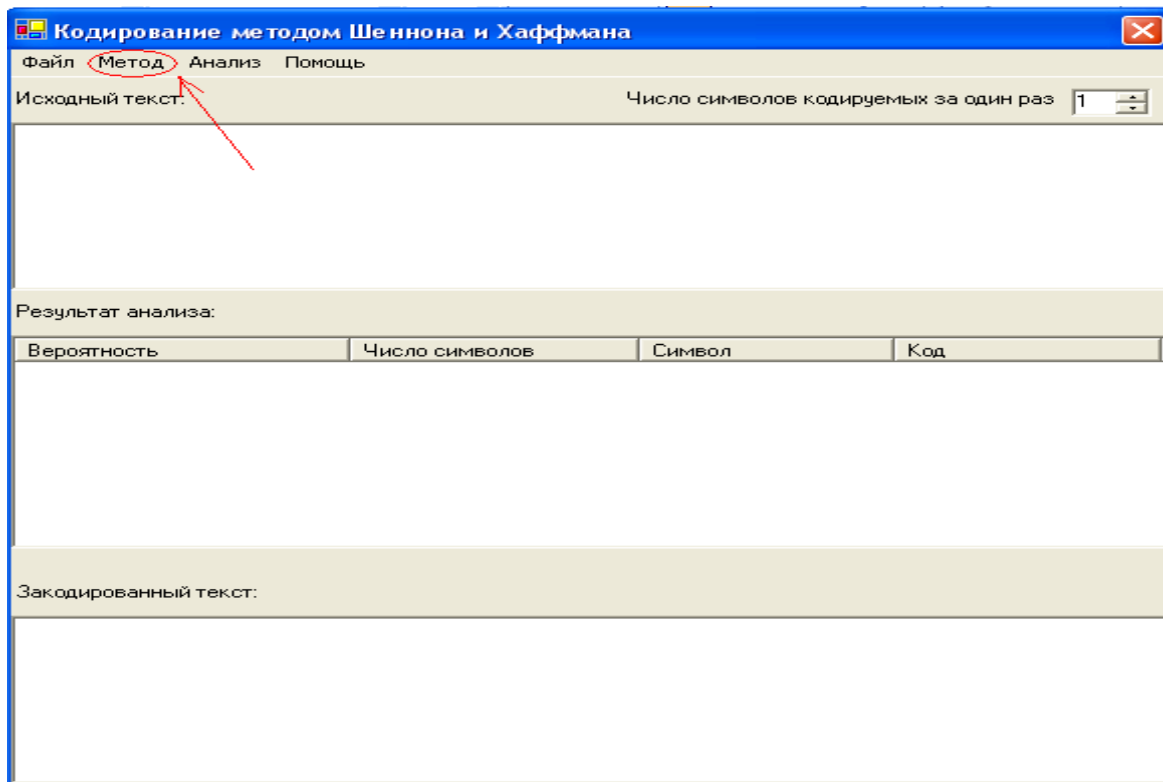
1. Description laboratory work

The work is performed on a personal computer using the *Shannon-Huffman.exe* program. The program is designed to demonstrate data compression methods using the Shannon-Fano and Huffman algorithms (Fig. 11.1).



Rice. 11. 1. Main program window

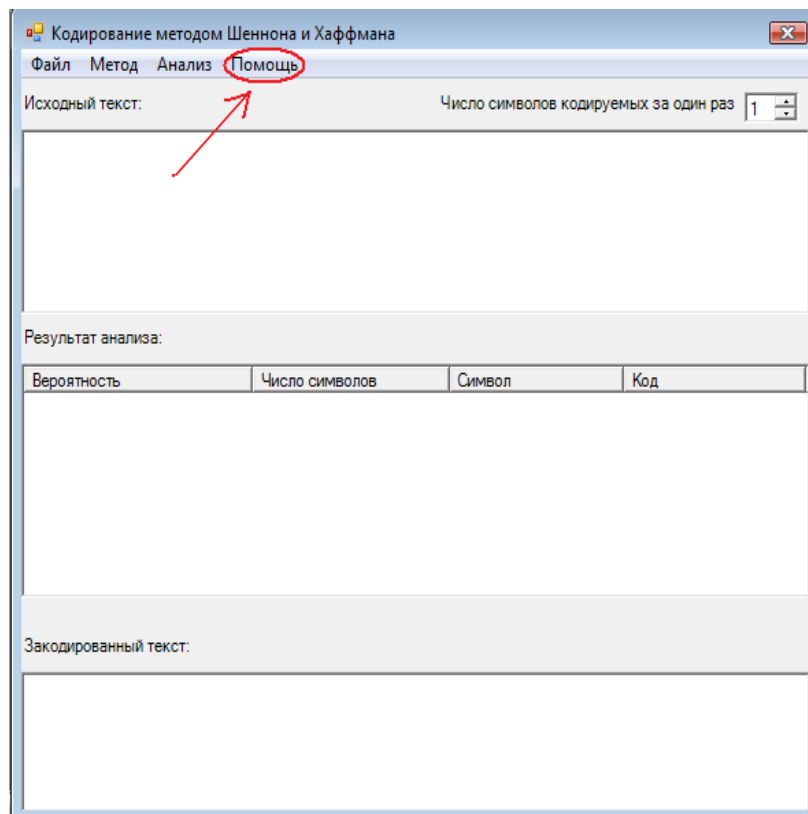
To work with the program, the user selects the required compression method, Fig. 11.2.



Rice. 11. 2. Choice method compression

Next, in the *Source Text program window* , a message of arbitrary length is written (or a message is loaded from a previously prepared file in *.txt format*). After this, it is necessary to specify the number of characters to be encoded at a time.

To calculate the probability of letters appearing in the entered message P , to determine the entropy of the message source H , the average number of symbols when encoding one letter of the message L , you must select the Analysis tab .



To determine the coding efficiency, the code redundancy (LH) is calculated.

To go out from this programs enough choose bookmark *File* And further *Exit*.

Example 1.

Original message *abbbvvggyydddeeeee* , consisting from symbols of the six-letter alphabet

$A = \{a, b, c, d, e\}$, we encode using the Huffman method. We determine the probability of the appearance of a symbol in the original message

$$P = n/N,$$

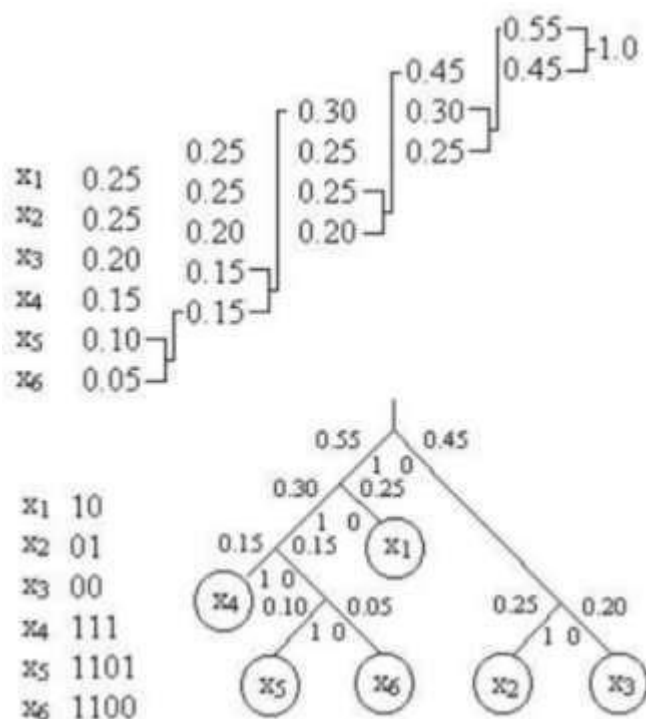
Where n - number repetitions symbol V message , N - message length .

We are writing out V column All symbols alphabet V in order decreasing probability of their appearance in the text (Table 11.1).

Table 11.1

symbol	number repetitions in this mess age puppies	probabi lity
<i>d</i>	5	0.25
<i>is</i>	5	0.25
<i>g</i>	4	0.20
<i>in</i>	3	0.15
<i>b</i>	2	0.10
<i>and</i>	1	0.05

By successively combining two symbols with the lowest probabilities of appearance into a new composite symbol, the probability of appearance of which is considered equal to the sum of the probabilities of the components its symbols, we will eventually construct a tree, each node of which has the total probability of all the nodes below it. We trace the path to each leaf of the tree, marking the direction to each node (for example, to the right - 0, to the left - 1).



Среднее количество символов на букву сообщения:

$$L = \sum_{i=1}^n P(i)n(i) = 2*2*0.25 + 2*0.20 + 3*0.15 + 4*(0.10+0.05) = 2.45,$$

где

$n(i)$ — количество знаков в кодовой комбинации i -го символа алфавита

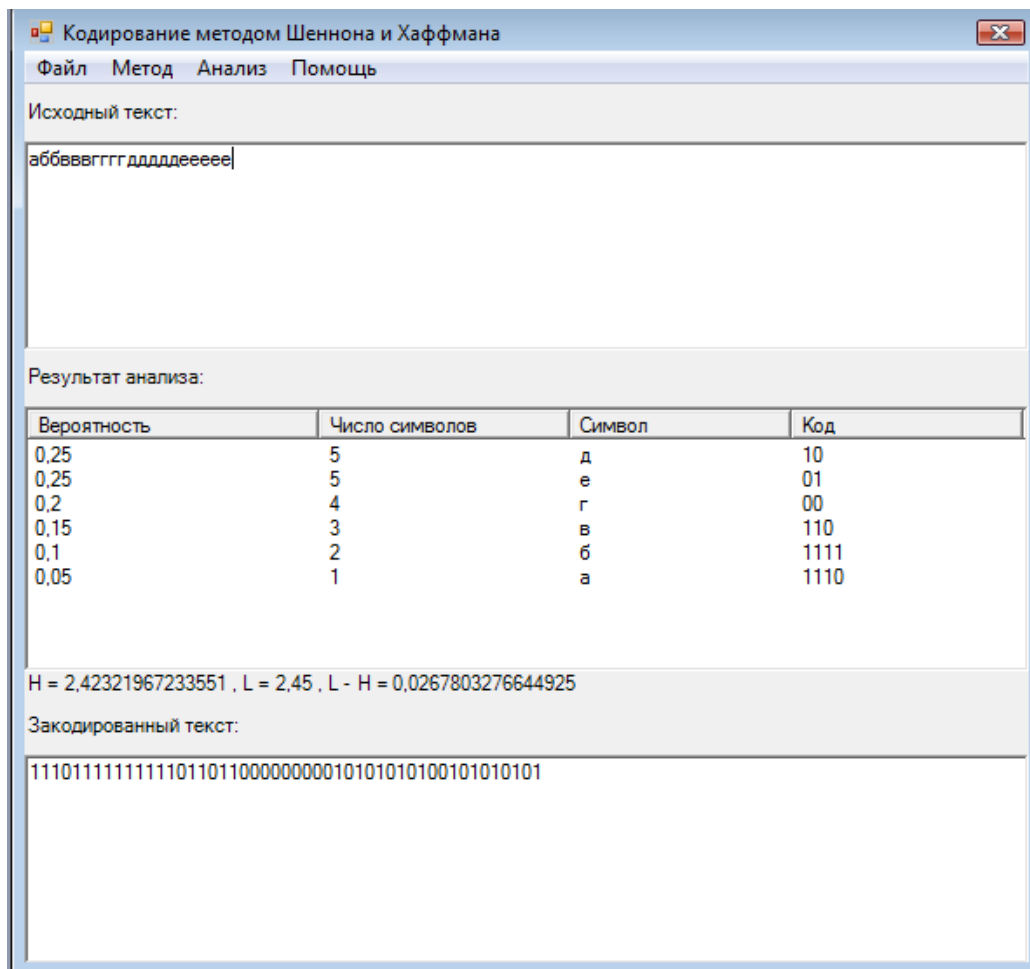
$P(i)$ — вероятность появления i -го символа алфавита.

Энтропия:

$$H = \sum_{i=1}^n P(i) \log \frac{1}{P(i)} = 2.425$$

Значение избыточности кода $(L-H) = 0.025$.

Let's compare data obtained With results work programs (Fig. 11.4)



Rice. 11. 4. Example coding

Example 2.

Let's encode the original message (see *Example 1*), but using Shannon's method. We write out all the symbols in a column in descending order of their probability of appearing in the text;

We sequentially divide the set of symbols into two subsets such that the sum of the probabilities of the appearance of symbols in one subset is approximately equal to the sum of the probabilities of the appearance of symbols in the other. For the lower subset, we assign "0" to each symbol; for the upper subset, we assign "1." Further partitions are repeated until all subsets consist of a single element (Table 11.2).

Table 11.2

symbol	number of repetitions in this message	probability	Code Shannon		
<i>d</i>	5	0.25	1	1	
<i>e</i>	5	0.25	1	0	
<i>G</i>	4	0.20	0	1	1
<i>V</i>	3	0.15	0	1	0
<i>b</i>	2	0.10	0	0	1
<i>A</i>	1	0.05	0	0	0

Average length received code will equal

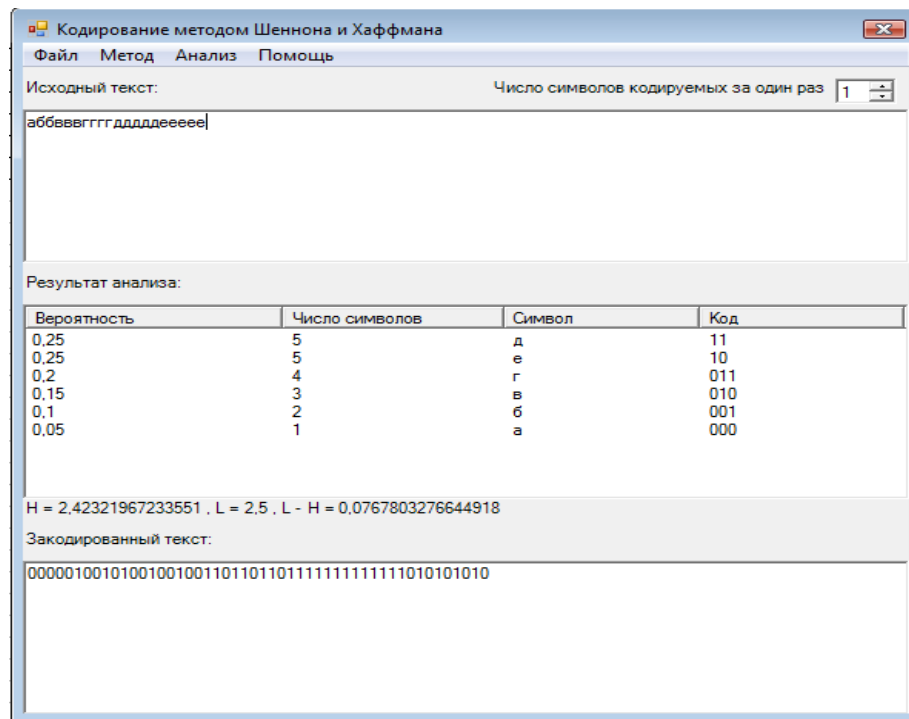
Average quantity symbols on letter messages:

$$L = \sum_{i=1}^n n(i) * P(i) = 0.25 * 2 + 0.25 * 2 + 0.2 * 3 + 0.15 * 3 + 0.1 * 3 + 0.05 * 3 = 2.5$$

Entropy:

$$H = - \sum_{i=1}^n P(i) \log \frac{1}{P(i)} = 2.425$$

We compare the obtained data with the results of the program (Fig. 11.5)



Rice. 11. 5. Coding example

2. Exercise

Find out more about the *Shannon-Huffman.exe* program , and consider examples of effective information encoding.

1. Run the *Shannon-Huffman.exe* program , which is designed to demonstrate data compression methods using the Shannon-Fano and Huffman algorithms.

2. Execute proposed tasks:

1). Number symbols alphabet $k = m$ (m – number options).

Compose such original message, to

- a) alphabet symbols were found in the message with equal probabilities ;
- b) alphabet symbols appeared in the message with different probabilities .

2). Enter arbitrary connected text on Russian language.

This could be a proverb, a poem, or any text. Using the program's output, analyze the alphabet of the input message: count the number of alphabetic characters, the entropy value H , and the average number of characters per character L for integer encoding.

3. Save screen forms in the report demonstrating the information compression process.

4. The laboratory report should contain the results of the analysis of the *Shannon-Huffman.exe* program during encoding; the algorithm for constructing the Huffman and Shannon code using knowledge on this topic; the results of the comparison of various encoding methods; conclusions about the efficiency of the compression methods used.

5. Bring V report answers on control questions, V in accordance with the option number indicated by the teacher.

Turn on V report O laboratory work completed tasks 1, 2,
3.

Options tasks

Task 1.

The message consists of a sequence of two letters A and B, the probabilities of the appearance of each of which do not depend on which was transmitted earlier, and are equal to 0.8 and 0.2, respectively. Perform encoding using the Shannon method: a) individual letters; b) blocks consisting of two-letter combinations; c) blocks consisting of three-letter combinations. Compare the resulting codes according to their effectiveness.

Exercise 2.

Compose a text that would match data given in Table 11.3.

Use the *Shannon-Huffman.exe* program to encode the text using the Huffman method.

Table 11.3

Number Option	probability	symbol	number of characters
1,5,7, 3,9,12,14,18,28	0.33333333	O	2
	0.16666667	G	1
	0.16666667	r	1
	0.16666667	d	1
	0.16666667	and	1
2,4,6,8,16,21 20,22,24, 30	0.25	is	2
	0.25	t	2
	0.125	about	1
	0.125	p	1
	0.125	r	1
	0.125	and	1
11,13,15, 10,17,19,23,25	0.25	r	2
	0.25	a	2
	0.25	c	2
	0.125	t	1
	0.125	e	1

Task 3.

For options A), B), C) (see Fig. 11.6 and Table 4), create a Huffman code. Calculate the average number of symbols on sign; of redundancy $(L - H)$; And relative redundancy received code $(L - H) / L$. Compare the obtained values with L , H , (LH) for the Shannon code, draw conclusions.

Δ)

Результат анализа:			
Вероятность	Число символов	Символ	Код
0,4375	7	а	11
0,1875	3	б	10
0,1875	3	г	01
0,125	2	в	001
0,0625	1	д	000

H = 2,05242128382936 , L = 2,1875 , L - H = 0,135078716170635

B)

Результат анализа:			
Вероятность	Число символов	Символ	Код
0,357142857142857	5	к	11
0,285714285714286	4	е	10
0,142857142857143	2	ж	011
0,142857142857143	2	з	010
0,0714285714285714	1	и	00

H = 2,1209520310264 , L = 2,28571428571429 , L - H = 0,164762254687883

IN)

Результат анализа:			
Вероятность	Число символов	Символ	Код
0,363636363636364	4	л	11
0,272727272727273	3	о	10
0,181818181818182	2	м	01
0,0909090909090909	1	н	001
0,0909090909090909	1	п	000

H = 2,11807820934971 , L = 2,18181818181818 , L - H = 0,0637399724684737

Rice. 11. 6. Options tasks

Table 11.4

Option number	Exercise
1,5,7, 3.9, 12,14,18, 28	<i>Exercise 3 . Option A).</i>

2,4,6,8, 20,22,23, 24.25 30	<i>Exercise 3 . Option B).</i>
11,13,15, 16 21,10,17, 19	<i>Exercise 3. Option IN).</i>

Number options	Control questions
1,5,7, 3,9,18,28	Which codes allow produce unambiguous decoding even without use dividing symbol? Give examples of such codes.
2,4,6,8, 20,22,24, 26:30	Name it conditions constructions optimal codes.
11,13,15, 10,17,19, 27	What is the purpose of effective codes, and which ones are you familiar with?
12,14,16 21,23.25, 29	List main methods compression information without losses.

MINISTRY SCIENCES AND HIGHER EDUCATION OF THE RUSSIAN
FEDERATION
FEDERAL STATE AUTONOMOUS EDUCATIONAL INSTITUTION OF HIGHER
EDUCATION
"NORTH CAUCASIAN FEDERAL UNIVERSITY"

METHODOLOGICAL INSTRUCTIONS

By organizations And conducting independent work on the discipline

Personal Data Protection in AI Systems

for students in the field of study
11.03.02 Infocommunication technologies And systems connections
Direction (profile)
"Intellectual processing data V infocommunication systems and networks"

Stavropol 2026

The guidelines for organizing students' independent work have been compiled in accordance with the requirements of the program. The discipline "Personal data protection in AI systems" for students majoring in 11.03.02 "Infocommunication technologies and communication systems." This teaching aid is dedicated to the planned academic work of students, carried out outside of class time. on assignment and with methodological teacher's manual, but without his direct participation (with partial direct participation of the teacher.

CONTENT

1. Independent Job How the most important form educational process	4
2. Goals And main tasks SRS	5
3. Types independent Works	6
4. Organization SRS	7
5. General recommendations By organizations independent works	8
6. Independent work of a student is a necessary part of development researcher	12
7. Methodological recommendations for students on individual forms independent work	13
8. Independent work of students in the conditions of a point-rating system systems training	20
9. Educational and methodological And informational security disciplines	23

1. Independent work as the most important form of the educational process

Independent work is the planned educational, educational and research, scientific research work of students, performed outside of class (classroom) time on assignment and with the methodological guidance of the teacher, but without his direct participation (with partial direct participation of the teacher, leaving the leading role to the students' work).

Independent work of students at a university is an important form of educational and scientific activity for students. Independent work Students' self-study plays a significant role in the rating technology of education. University education includes two parts, practically equal in scope and mutual influence: the learning process and self-study. Therefore, independent work should become an effective and targeted effort for the student.

The concept for modernizing Russian education defines the main objectives of vocational education as "the preparation of a qualified worker of the appropriate level and profile, competitive in the labor market, competent, responsible, fluent in their profession and oriented in related fields of activity, capable of effective work in their specialty at the level of world standards, ready for continuous professional growth, social and professional mobility."

The solution to these problems is impossible without increasing the role of students' independent work on educational material, strengthening the responsibility of teachers for developing independent work skills, stimulating students' professional growth, and fostering creativity and initiative.

Society places a wide range of demands on modern professionals, among which a significant factor is the ability of graduates to independently acquire knowledge from various sources, systematize the information obtained, and assess a specific financial situation. This ability is developed throughout the entire training period through students' participation in practical classes, completing assignments and tests, writing term papers, and final qualifying theses. Moreover, students' independent work plays a crucial role throughout the entire educational process.

Students' independent work options are varied. The course "Teletraffic in Multiservice Networks" offers the following:

Usage material educational and methodological complex disciplines

On first stage necessary get acquainted With mandatory the minimum content of the basic educational program for the subject being studied The course. Next, you need to study the course syllabus, which covers the content of the lecture course topics, the relationship between lecture topics and laboratory classes, and the topics and types of independent work. Specific reporting forms are provided for each type of independent work.

Job With literature

To successfully master the course, you must independently and thoroughly study the presented topics using the recommended sources of information:

Item No.	Types independent work	Recommended sources information (No. source)			
		Main	Additional- telny	Methodological	Internet- resources
1	Independent studying literature	1, 2	1, 2	1	1, 2, 3

Independent work introduces students to scientific creativity, searching for and solving current modern problems.

2. Goals And main tasks SRS

The primary goal of organizing and implementing independent work should align with the goal of student education—the preparation of a bachelor's degree holder with a higher education degree. When organizing independent work, developing the ability to work independently to acquire knowledge, skills, and the ability to organize educational and research activities is an important and necessary condition.

The goal of students' independent work is to acquire fundamental knowledge, professional skills, and relevant professional abilities, as well as experience in creative and research activities. Independent work promotes the development of students' independence, responsibility, self-discipline, and a creative approach to solving academic and professional problems.

Tasks SRS are:

- systematization and consolidation of the acquired theoretical knowledge and practical skills of students;

- deepening And extension theoretical knowledge;
- development of skills in using regulatory, legal, reference documentation and specialized literature;
- development of students' cognitive abilities and activity: creative initiative, independence, responsibility and organization;
- development of independent thinking, abilities for self-development, self-improvement and self-realization;
- development research skills;
- use of material collected and obtained during independent study at seminars, in practical and laboratory classes, when writing coursework and final qualifying papers, for effective preparation for final tests and exams.

3. Types independent work

In the educational process of higher professional educational institution, two types of independent work are distinguished

– classroom, under the guidance of a teacher, and extracurricular. The close relationship between these types of work ensures differentiation and the effectiveness of their results and depends on the organization, content, and logic of the educational process (interdisciplinary connections, advanced knowledge, etc.):

Independent classroom work on the subject is carried out during classes under the direct supervision of the teacher and according to his/her instructions.

Extracurricular independent work is performed by a student on the teacher's instructions, but without his or her direct participation.

The main types of independent work of students without the participation of a teacher in this discipline are:

- formation and assimilation of the content of lecture notes based on the educational literature recommended by the lecturer, including informational educational resources (electronic textbooks, electronic libraries etc.);
- Preparation To laboratory works, their design.

The main ones species independent work students With The participation of teachers is:

- current consultations;
- reception And protection laboratory works (in time conducting l/r).

4. Organization SRS

The methodology for organizing students' independent work depends on the structure, nature, and characteristics of the discipline "Teletraffic of Multiservice Networks," the number of hours for its study, the type of assignments for students' independent work, the individual qualities of students, and the conditions of educational activities.

The process of organizing students' independent work includes the following stages:

- preparatory (defining goals, drawing up a program, preparing methodological support, preparing equipment);
- basic (implementation of the program, use of techniques for searching for information, assimilation, processing, application, transfer of knowledge, recording of results, self-organization of the work process);
- final (assessment of significance and analysis of results, their systematization, assessment of the effectiveness of the program and work methods, conclusions on areas of labor optimization).

Activity students By formation And development skills of independent educational work.

In the process of independent work, the student acquires the skills of self-organization, self-control, self-management, self-reflection and becomes an active, independent subject of educational activity.

Carrying out independent work under control the teacher the student must:

- master the minimum content assigned to students for independent work and proposed by the teacher in accordance with the educational standards of higher education;
- plan independent work in accordance with the independent work schedule proposed by the teacher;
- The student must carry out independent work in the organizational forms provided for by the curriculum and the teacher's work program;
- perform independent work and report on it results in accordance with the schedule for presenting results, types and deadlines for reporting on students' independent work.

student Maybe:

in excess of that proposed by the teacher (with justification and (in agreement with him) and the minimum required content determined by the program for this discipline:

- independently determine the level (depth) of elaboration of the content of the material;
- suggest topics And questions For independent developments;
- within the general schedule for completing independent work, propose a reasonable individual schedule for completing and reporting on the results of independent work;
- propose your own options for organizational forms of independent work;
- use for independent work methodological aids, teaching aids, and developments in addition to the list proposed by the teacher;
- use not only control, but also self-control of the results of independent work in accordance with the methods of self-control proposed by the teacher or chosen independently.

Students' independent work should have a significant impact on the development of their future bachelor's degree students. It is planned by the student themselves. Each student determines their own work schedule and the amount of effort expended on mastering the academic content in each subject. They complete their extracurricular work according to their individual plan, depending on their preparation, time, and other circumstances.

5. General recommendations By organizations independent work

The primary form of student independent work is studying lecture notes, supplementing them, referring to recommended literature, and actively participating in laboratory classes. However, to ensure successful learning and its intensification, the following subjective factors must be considered:

1. Knowledge of the school curriculum and a solid system of gaps necessary for mastering core university courses are essential. This is especially important for mathematical subjects. It's important to distinguish between gaps in knowledge that hinder the assimilation of new material and limited abilities. By investing effort in overcoming these gaps, a student will ensure adequate academic performance and gain confidence in their abilities.

2. Availability skills, skills mental labor:

A) skill take notes on lectures And at work With book;

b) mastery of logical operations: comparison, analysis, synthesis, generalization, definition of concepts, rules of systematization and classification.

3. The specific features of cognitive mental processes include attention, memory, speech, observation, intellect, and thinking. Underdevelopment of any of these can become a serious obstacle to learning.

4. Good performance, which is ensured by normal physical condition. After all, serious study is a complex, multifaceted, and varied undertaking. Learning outcomes are assessed not by the quantity of information imparted, but by the quality of its assimilation, the ability to use it, and the development of one's capacity for further independent learning.

5. Matching your chosen activity or profession to your individual abilities is crucial. It's essential to develop the ability to self-regulate your emotional state and eliminate circumstances that disrupt your professional spirit and interfere with your intended work.

6. Mastering an optimal work style that ensures success in your work. Alternating work and breaks, rest periods, individually tailored sleep patterns, preference for evening or morning classes, stress tolerance during exams, and specific preparation techniques.

7. The level of demands on oneself, determined by the established self-esteem.

An adequate assessment of knowledge, strengths, and weaknesses is an important component of human self-organization; without it, successful work on managing one's behavior and activities is impossible.

One of the main features of higher education is that constant external control is replaced by self-control; the active role in learning belongs not so much to the teacher as to the student.

Knowing the basic methods of scientific organization of mental work, one can achieve the best results with the least expenditure of time, resources and labor effort.

The effectiveness of assimilation of incoming information depends on a person's performance at any given moment of his activity.

Work capacity is a person's ability to work at a high level of intensity for a certain period of time. A distinction is made between internal and external factors of work capacity.

Internal factors of performance include intellectual abilities, willpower, and health status.

TO external:

- organization worker places, mode labor And rest;
- level of work organization - the ability to obtain information and use it;
- magnitude mental loads.

The outstanding Russian physiologist N. E. Vvedensky identified the following conditions for the productivity of mental activity:

- in any work need to enter gradually;
- dimensionality and rhythm work. Different people inherent more or less variable work pace;
- habitual subsequence And systematicity activities;
- correct alternation labor And rest.

Rest doesn't necessarily require complete inactivity; it can be achieved by simply changing tasks. Work capacity fluctuates throughout the day. The most productive *time is morning (8:00 a.m. to 2:00 p.m.)*, with peak productivity occurring between 10:00 a.m. and 1:00 p.m., followed by *afternoon (4:00 p.m. to 7:00 p.m.)* and *evening (8:00 p.m. to midnight)*. It's best to study very difficult material at the beginning of each work period (preferably morning) after a good rest. Breaks of 10-15 minutes are needed after 1-1.5 hours of work, and after 3-4 hours of work, rest should be longer – about an hour.

An integral part of the scientific organization of mental work is the mastery of the technique of mental work.

The time available to a student to complete the curriculum consists of two components: one of them is classroom work at the university One is scheduled class time, and the other is independent work outside of class. Assignments and materials for independent work are provided during scheduled classes. During these classes, the instructor monitors independent work and provides assistance to students in organizing their work properly.

Independent study requires intense mental effort, which must not only be properly organized but also stimulated. It's crucial to be able to maintain sustained attention to the material being studied. Developing attention requires significant willpower. Therefore, if a student notices they are frequently distracted during independent study, they need to force themselves to focus. This procedure necessary to do constantly, So How This

is a training of attention. Sustained attention appears when a person approaches a task with interest.

It's important to organize your activities properly: 50 minutes of work, 5-10 minutes of breaks; after 3 hours of work, take a 20-25 minute break. Otherwise, increasing fatigue will lead to a loss of focus. Regular physical exercise is a very important factor in improving mental performance. Organizing active recreation involves alternating mental and physical activity, which fully restores a person's performance.

6. Independent work of a student is a necessary part of becoming a researcher

The progress of science and technology, information technology leads to a significant increase in scientific information, which places higher demands not only on the moral and ethical qualities of a person, but also, in particular, constantly increasing demands in the field of education - updating, modernization of general and professional knowledge and skills of a specialist.

All education should be a dynamic process, inherent to the individual and continuing throughout their life. Mastering scientific thought and the language of science is a necessary component in the self-organization of a future specialist researcher. This This refers not so much to the accumulation of knowledge as to the mastery of scientifically proven methods for acquiring it. This, generally speaking, is the main task of a university.

The unique nature of the university educational process, in which independent student work is increasingly emphasized, is that it seems to be the final and most appropriate link in achieving this goal. During university studies, students develop a style and skills for learning (cognitive) activity, the rational nature of which will facilitate the continuous renewal of knowledge in highly qualified university graduates.

However, there are certain difficulties along the way, particularly the student's transition from the synthetic learning process in secondary school to the analytical one in higher education. This is due both to the new content of instruction (the expansion of general education and the deepening of professional training) and to new, previously unknown forms of instruction (lectures, seminars, laboratory classes And etc.). Student receives Not Not only the knowledge provided by the curriculum and textbooks, but they must also become familiar with the methods of acquiring knowledge so as to be able to evaluate what we know, where we know it from, and how we achieved it. All of this is achieved through independent work.

This is because independently acquired knowledge is more immediate, becoming personal property and a motivation for behavior, developing intellectual traits, attention, observation, critical thinking, and the ability to evaluate. The role of the teacher is primarily to guide the accumulation of knowledge (for first-year students), and in subsequent years of study, in senior years, in a joint effort. establishment problems And care O independent searches

students, as well as monitoring their activities. It should be noted that one should not limit oneself to acquiring only the knowledge provided by the program of the discipline being studied; one must constantly deepen the acquired knowledge, focusing it on a narrow, specific area that corresponds to the student's interests. In-depth study of all The subjects provided for by the program are possible in practice, and good organization of work allows saving time, which creates conditions for a deep, systematic, and interested study of the topic independently chosen by the student.

Of course, all the advice, examples, recommendations in this area, The recommendations provided by a teacher, certain publications, or other sources do not guarantee any success without personal action. They do not provide ready-made recipes, but rather should encourage an analysis of one's own work, its goals, and organization in accordance with individual characteristics. Taking into account personal capabilities, existing living and working conditions, and skills, using these recommendations, it is possible to develop an individually valid set of methods and techniques, find one's own style, or refine it. After studying specific material, one has the time to evaluate its significance, suitability, and potential for application, ultimately ensuring success in one's studies and future professional activities.

7. Methodological recommendations for students on certain forms of independent work

The university education system requires students to have significantly greater independence in planning and organizing their activities. This can be quite challenging for a recent high school student: while at school, daily teacher supervision forced students to constantly and systematically prepare for classes, at university, the question of their knowledge level only really arises during exam periods. This situation leads some to the temptation to devote the entire semester to free time ("I'll learn it when I need to!"), and when exam time arrives, the material to be mastered is so overwhelming that no memory can cope with it in the remaining time.

Job With literature.

When working with books, you need to select literature, learn to read it correctly, and keep records. To select literature, the library uses alphabetical and systematic catalogs.

Important remember, What rational skills work With book - This always a big saver of time and effort.

The instructor teaching the course recommends the appropriate selection of textbooks. Required literature may also be listed in the course's teaching materials.

When studying the material in a textbook, you should move on to the next question only after correctly understanding the previous one, describing on paper all the calculations and computations (including those that are omitted from the textbook or given in the lecture for independent derivation).

When studying any discipline, independent individual work plays a large and important role.

Particular attention should be paid to defining the course's key concepts. Students should thoroughly analyze examples that illustrate these definitions and be able to construct similar examples independently. It's important to develop a clear understanding of what you're learning. Taking supporting notes is helpful. When studying the material from the textbook, it's helpful to supplement your lecture notes in the designated margins in your notebook. You should also note any questions you've identified for consultation with the instructor.

The findings obtained as a result of the study are recommended in Highlight the notes in your notes so that they are better remembered when you reread them.

Experience shows that many students find it helpful to create a reference sheet containing the most important and most frequently used signals. The formulas and concepts used. This sheet helps students remember formulas and key points from the lecture, and can also serve as a constant reference for students.

There are two types of reading: primary and secondary. *Primary* reading is attentive, unhurried reading, allowing you to pause at difficult passages. Afterward, there shouldn't be a single unclear point. The content may not always be clear after the primary reading.

The goal of *secondary* reading is to fully grasp the meaning of the whole (by counting this reading may not be the second, but the third or fourth).

Rules independent work With literature.

As already noted, independent work with textbooks and books (As well as independent theoretical research of the problems outlined by the instructor in lectures) is a crucial condition for developing a scientific approach to learning. The main tips here can be summarized as follows:

- Compose scroll books, With which To you follows познакомиться; “Don’t try to remember everything that you won’t need in the near future,” advises G. Selye to a student and young scientist, “just remember where you can find it” (Selye, 1987. P. 325).

- The list itself must be systematized (which is necessary For seminars, What For exams, What will come in handy For writing term papers and theses, and what interests you outside the framework of official academic activities, that is, what can expand your general culture...).

- It is essential to write out all the output data for each book (when writing term papers and theses, this will save a lot of time).

- Figure out for yourself which books (or which chapters of books) you should read more carefully, and which ones you should just skim.

- When compiling your reading lists, you should consult with your professors and supervisors (or even with more prepared and knowledgeable fellow students), who will help you better understand what you should pay more attention to and what you shouldn't waste your time on at all...

- Naturally, all the books, textbooks and articles read should be taken notes, But This Not means, What I need to take notes "All "in a row": you can briefly write out the author's main ideas and sometimes provide the most striking and illustrative quotes (with page numbers).

- If book - Yours own, That allowed do on fields books, make brief notes or at the end of the book, on the blank pages, simply make your own “subject index”, where you note the thoughts that are most interesting to you and be sure to indicate the pages in the author’s text (this is a very good advice, allowing save time And fast find "the chosen ones" places V the most different books).

- If you haven't worked much with scientific papers before, literature, then you should develop the ability to “perceive” complex texts; the best technique for this is to learn to “read slowly,” when you understand every word you read (and if the word is unfamiliar, then either with the help of a dictionary, or With help teacher Necessarily his to know), And This Maybe take up

a considerable amount of time (for some, up to several weeks or even months); experience shows that after this, the student, by some miracle, begins to literally devour books and can almost see "through the covers" whether the work is worthwhile or not...

- “Either read or skim the material, but don’t try to read quickly... If the text interests me, then reading, thinking, and even fantasizing By to this about merge V single process, V That time “as forced speed reading not only does not contribute to the quality of reading, but also does not bring the feeling of satisfaction that we get from thinking about what we have read,” advises G. Selye (Selye, 1987. – Pp. 325-326).

- Another effective way to optimize your exposure to scientific literature is to become fascinated by an idea and then view all books from the perspective of that idea. In this case, the student (or young scientist) will be searching for arguments for or against the idea in question, while simultaneously engaging with the authors of these ideas. books By about their own ideas And reflections... Problem only in how to find “your” idea...

Reading scientific text is part of cognitive activity. Its purpose is to extract necessary information from the text. From how conscious the reader is of his own internal attitude when addressing the printed word (to find the necessary information, to assimilate information in whole or in part, critically analyze the material, etc.) the effectiveness of the action taken largely depends .

They highlight *four main installations V reading scientific text* :

1. information retrieval (the task is to find and highlight the desired information)

2. assimilative (the reader's efforts are aimed at fully comprehending and remembering both the information presented by the author and the entire logic of his reasoning)

3. analytical-critical (the reader strives to critically comprehend the material, having analyzed it, and determined his attitude toward it)

4. creative (creates in the reader a readiness in one form or another - as a starting point for his own reasoning, as an image for action by analogy, etc. - to use the author's judgments, his train of thought, the result of observation, the developed methodology, to supplement them, to subject them to new testing).

With the presence of various settings for accessing scientific text The existence of several *types of reading* is also connected :

1. bibliographic – viewing catalog cards, recommendation lists, summary lists of journals and articles for the year, etc.;

2. browsing - used to search for materials containing the necessary information, usually resorted to immediately after working with bibliographies and catalogues, as a result of such browsing the reader determines which sources will be used in further work;

3. introductory - implies a continuous, fairly detailed reading of selected articles, chapters, individual pages, the goal is to become familiar with the nature of the information, find out what issues the author has raised for consideration, and sort the material;

4. studying - suggests thorough development material; V During such reading, the reader's trust in the author is demonstrated, he is willing to accept the information presented, and he is driven to fully understand the material;

5. analytical-critical and creative reading are two types of reading that are close to each other in that they participate in solving research problems tasks. The first involves a focused critical analysis of both the information itself and the author's methods of obtaining and presenting it; the second involves searching for those judgments and facts about which, or in connection with which, the reader considers it necessary to express their own thoughts.

Of all the types of reading discussed, exploratory reading is the most important for students—it allows them to accumulate knowledge in various fields while working with educational literature. That's why this type of reading should be mastered first in the context of educational activities. queue. Moreover, mastering this type of reading develops basic techniques that increase the effectiveness of working with scientific text.

Main types systematized records read:

1. Annotation is an extremely brief, coherent description of a book (article) viewed or read, its content, sources, nature and purpose;

2. Planning is a brief logical organization of the text, revealing the content and structure of the material being studied;

3. Thesis is a concise reproduction of the author's main assertions without using factual material;

4. Quoting is the verbatim copying of excerpts and extracts from the text that most significantly reflect a particular idea of the author;

5. Note-taking is a brief and sequential presentation of the content of what has been read.

A summary is a complex way of presenting the contents of a book or article in a logical sequence. A summary integrates previous notes and allows for a comprehensive understanding of the book or article's content. Therefore, the ability to create an outline, make notes, take excerpts, and other notes determines the technique of summary writing.

Methodological recommendations By compilation summary:

1. Read the text carefully. Look up any unfamiliar words in reference books. When taking notes, be sure to include reference information in the margins.

2. Select The main thing, make up plan.

3. Briefly formulate the main points of the text, note the author's argumentation.

4. Take notes on the material, strictly following the outline. When taking notes, try to express your thoughts in your own words. Keep your notes clear and concise.

5. Write quotes correctly. When quoting, keep in mind the brevity and significance of the idea.

It's advisable to include not only thesis statements but also their evidence in the summary. When formatting the summary, strive for the conciseness of each sentence. The author's thoughts should be presented concisely, with care taken to maintain style and expressiveness. The number of additional elements in the summary should be logical. To be justified, entries must be arranged in a specific sequence that corresponds to the logical structure of the work. Margins must be left for clarification and addition.

Mastering note-taking skills requires dedication and independent work on a daily basis from the student.

Laboratory classes.

It should be emphasized that only after mastering the lecture material from a certain point of view (namely, from the one from which it is presented in lectures) will it be consolidated in practical classes both as a result of discussion and analysis of the lecture material, and through the implementation of tasks laboratory work, delivered tasks. At these

In this way, the student will not only master the material well, but will also learn to apply it in practice, and will also receive additional incentive (and this is very important) to actively study the lecture.

When completing assignments independently, each step of the solution should be justified based on the theoretical principles of the course. If the student sees several possible solutions to a problem (task), they should be compared and the most efficient one selected. It is helpful to create a brief plan for solving the problem (task) before beginning calculations. Solutions to problems or examples should be presented in detail, with calculations arranged in strict order, separating auxiliary calculations from the main ones. Solutions should be accompanied by comments, diagrams, drawings, and illustrations, if necessary.

It's important to remember that each learning task should be completed to the final logical answer required by the conditions, and, if possible, with a conclusion. The resulting answer should be verified using methods appropriate to the nature of the problem. It's also useful (if possible) to solve the problem in several ways and compare the results. For example, tasks are completed using NetCracker Professional 4.1, then analytically and the results are compared to substantiate the outcome of the task. Laboratory work. This type of task should be continued until solid solving skills are acquired.

Self-test.

After studying a certain topic from notes in the notes and textbook, as well as solving a sufficient number of relevant tasks in laboratory classes and independently by the student. It is recommended that, using a sheet of supporting materials, you reproduce from memory the definitions, derivations of formulas, formulations of the main provisions and evidence.

If necessary, you need to carefully review the material again .

Sometimes, a student's lack of understanding of a particular topic only becomes apparent upon further study. In this case, it's necessary to go back and review the poorly understood material. An important criterion for mastering theoretical material is the ability to solve problems or pass tests on the material covered. However, it's important to remember that a correct solution to a problem can result from applying rote memorized formulas without understanding the essence of the theoretical principles.

Consultations

If in the process of independent work on studying theoretical material or when completing assignments from a student. If questions arise that you can't resolve on your own, you should contact your instructor for clarification or guidance. In your questions, students should clearly explain the difficulty and the nature of the difficulty. You should also seek advice if you have any doubts about your answers to the self-assessment questions.

8. Independent work of students in the context of a point-rating system of education

The rating system of education involves multi-point student assessment, but this is not a simple transition from the five-point scale. Rather, it allows for the objective reflection in points of a broader range of assessment of students' individual abilities and their efforts expended on completing a particular type of independent work. There is ample scope for creating a set of differentiated individual assignments, each with its own "value." A properly organized rating system of education allows for this from the very beginning. began to move away from the five-point grading system and only return to it at the end of the assessment, when the points earned by students are converted into standard grades (excellent, good, satisfactory, unsatisfactory). In addition, the rating system includes additional incentive points for originality and novel approaches to completing independent work assignments or solving scientific problems. Students have the opportunity to improve their academic rating by participating in extracurricular activities (participation in Olympiads, conferences; completing individual creative assignments, papers; participating in a science club, etc.). Students who are slow to submit assignments on time may receive negative marks. However, individual students are encouraged to complete the program more quickly. For example, if a student is ready to take a test or write an independent assignment ahead of the group, they may receive additional marks.

The rating system is a regular monitoring of the quality of acquisition of knowledge and skills in the educational process, the fulfillment of the planned volume of independent work work. Management multi-point systems assessments allows,

On the one hand, to reflect the individual characteristics of students in a point range, and on the other hand, to objectively evaluate in points the students' efforts expended on completing individual types of work. Thus, each type educational activities acquires my "price". It turns out, What

The "value" of work performed flawlessly by a student is a quantitative measure of the quality of his learning based on the totality of the educational material he studied that was necessary for the successful completion of the assignment.

The maximum possible score for the entire ongoing assessment is set at 55. A current assessment is considered passed if the student receives at least 60% of the maximum score established for that assessment. The rating score assigned to a student for a current assessment completed within the deadlines set in the assessment schedule is determined as follows:

Level of completion of the control task	Rating score (as a percentage of the maximum score) for the test task)
Great	100
Good	80
Satisfactory	60
Unsatisfactory	0

The final grade for a course studied in one semester is determined by the sum of points scored for work during the semester.

Scale for converting the rating score for a discipline into a grade on a 5-point system

Rating score by discipline	Grade By 5-point system
88 – 100	Great
72 – 87	Fine
53 – 71	Satisfactorily
< 53	Unsatisfactory

The developed scale for converting a discipline rating into a final five-point grade is accessible and easily calculated by both the teacher and the student.

At use rating systems:

- the main emphasis is placed on organizing active types of educational activities; students' activity goes towards creative understanding of the proposed tasks;
- in the relationship between the teacher and students there is cooperation and co-creation, there is psychological and practical

the teacher's readiness to recognize the individual uniqueness of each student's "I-concept";

- a variety of stimulating, emotional-regulating, guiding and organizing intervention techniques is assumed (if necessary) the teacher in the independent work of students;

- The teacher acts as a manager and director of learning, ready to offer students the minimum necessary set of learning tools, and not just convey educational information; the student acts as a subject of activity alongside the teacher, and the development of his or her individuality is one of the main educational goals;

- educational information is used as a means of organizing educational activities, and not as a goal of learning.

The rating system of education ensures the greatest informational, procedural and creative productivity of independent cognitive activity of students, provided that it is implemented through the technologies of personally-oriented learning (problem-based, dialogic, discussion, heuristic, game and other educational technologies).

Most students respond positively to this system of tracking their progress, noting that the rating system facilitates an even distribution of their efforts throughout the semester, improves their retention of course material, and ensures systematic work without the stress of rushing through exams. The large number of diverse assignments offered for independent study and the various grading scales allow students to monitor their progress and, if desired, always have the opportunity to improve their rating (by completing additional independent work) without waiting for an exam. Organizing the learning process within the rating system, using a variety of independent work options, results in higher student learning outcomes compared to traditional university education .

Using a rating system allows for more consistent student work throughout the semester and also enhances students' cognitive activity by stimulating their creativity. Direct testing is highly effective. V process training, at independent work

Students. In this case, the student tests their knowledge themselves. If they don't answer the test question immediately, they receive a hint explaining the logic behind it and complete it a second time.

It is also worth noting that automated teaching and teaching-control systems are increasingly penetrating the educational process, allowing students to independently study a particular subject and simultaneously monitor the level of assimilation of the material.

9. Educational, methodological and informational support of the discipline

For this subject, the teacher recommends the following literature:

IN quality basic literature are considered:

1. Iversen V.B. Development of Teletraffic and Network Planning Electronic resource: study guide / V.B. Iversen. - Development of Teletraffic and Network Planning, 2019-12-20. - M.: Internet University of Information Technologies (INTUIT), 2016. - 625 p. - The book is in the basic version of the IPRbooks electronic library system.

2. Reshetnikova I.V. Teletraffic Theory Electronic resource / Reshetnikova I.V.: study guide. - Rostov-on-Don: RGUPS, 2019. - 80 p. - ISBN 978-5-88814-923- 2.

IN quality additional literature:

1. Fundamentals of infocommunication technologies: teletraffic theory study guide / E.D. Bychkov, V.A. Maistrenko, O.N. Kovalenko, D.N. Kovalenko; under ed. V.A. Maistrenko; Ministry of Education and Science Russia; Omsk State Technical University. - Omsk: Omsk State Technical University Publishing House, 2017. - 156 With. : graph., schemes. - <http://biblioclub.ru/> . - Bibliography: With. 130-131. - ISBN 978-5-8149-2433- 9.

2. Bratchenko N.Yu. Teletraffic Theory Electronic resource: study guide / N.Yu. Bratchenko. - Stavropol: North Caucasus Federal University, 2014. - 177 p. - The book is in the basic version of the electronic library system IPRbooks.

Students Also can take advantage of methodological literature:

Teletraffic theory: a teaching aid: Training area 210700.62 (11.03.02)– Infocommunication technologies And

communication systems. Training profile "Communication networks and switching

systems" / author-compiler N. Yu. Bratchenko; FGAOU HPE North-Caucasian Federal University - Stavropol: NCFU, 2014. - 55 p. : ill.

IN quality Internet resources are offered next sources:

1. <http://biblioclub.ru>
2. <http://www.iprbookshop.ru>
3. <http://catalog.ncfu.ru/catalog/ncfu>

To independently master the theoretical and practical material, you will need the following software:

1. Operating room system families Windows Microsoft.
2. MathCad.
3. MS Office 2010.
4. MatLab.
5. Net Cracker 4.1.
6. MS Power Point.