

MINISTRY OF SCIENCE AND HIGHER EDUCATION
OF THE RUSSIAN FEDERATION
Federal State Autonomous Educational Institution
of Higher Education
“Kazan (Volga Region) Federal University”

Institute of Computational Mathematics and Information Technologies

APPROVED BY

Vice-Rector for Education

_____ E.A. Turilova

_____ 2025

**PROGRAM OF ENTRANCE EXAMINATION
IN INFORMATICS AND ICT**



Entrance examination program approval sheet

The program is developed by:

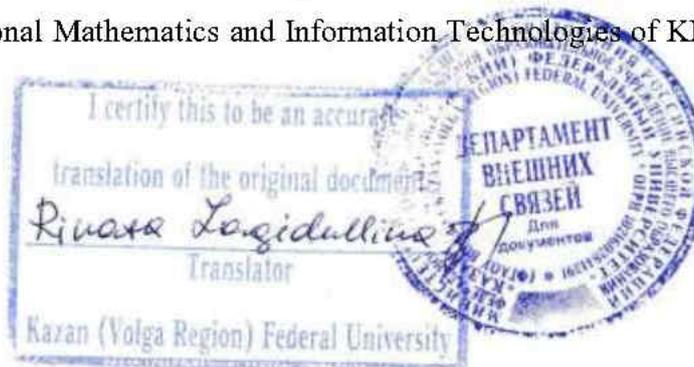
the Head of the Information Systems Department F. M. Gafarov

Chairperson of the Examination Commission F. M. Gafarov

The program of the entrance examination was discussed and approved at the meeting of the Information Systems Department of the Institute of Computational Mathematics and Information Technologies of KFU. Minutes No. 1 of September 3, 2025

By the decision of the Teaching and Methodical Committee of the Institute of Computational Mathematics and Information Technologies, the program of the entrance examination is recommended for approval by the Academic Council. Minutes No. 2 of September 19, 2025

The program of the entrance examination is approved at the meeting of the Academic Council of the Institute of Computational Mathematics and Information Technologies of KFU. Minutes No. 2 of September 29, 2025



Content

Part I. Introduction

- 1.1 Purpose and objectives of the entrance examination
- 1.2 General requirements for the organization of the entrance examination
- 1.3 Description of the format of the entrance examination
- 1.4 Duration of the entrance examination
- 1.5 Assessment criteria for the entrance examination

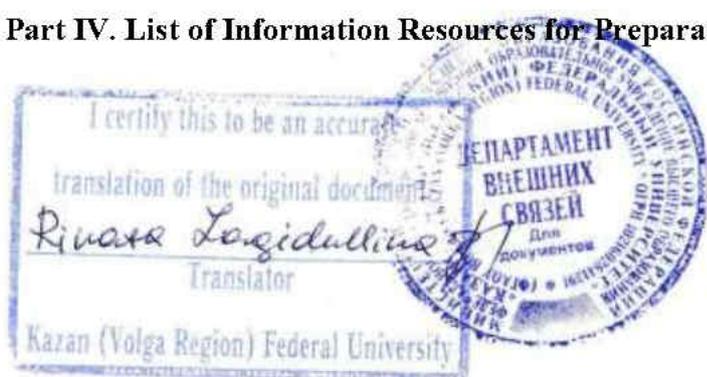
Part II. Program Content

- 2.1. Content of the test sections

Part III. Assessment materials

- 3.1. Instruction on task completion
- 3.2. Sample tasks

Part IV. List of Information Resources for Preparation for the Entrance Examination



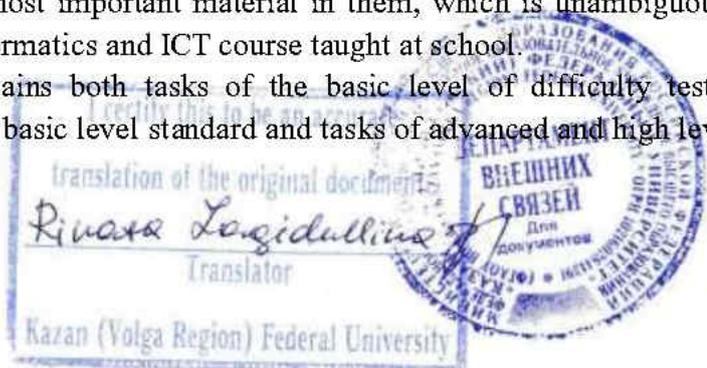
Section 1. Introduction

- 1.1. Purposes and objectives of the entrance examinations. The purpose of the entrance examination is to objectively assess the level of education of applicants in accordance with the requirements of bachelor's programs. The objectives of the entrance examinations are:
 - to determine the level of knowledge, skills and abilities of applicants in specialty-related subjects required for study under a bachelor-level program.
 - evaluate the intellectual and cognitive abilities of applicants, including logical thinking, analytical skills, the ability to work with information and apply knowledge in practice.
 - to ensure transparency, objectivity and equal conditions for all applicants when assessing their academic qualifications.
 - to contribute to the formation of the student population that holds sufficient potential for successful completion of a bachelor-level program and further professional development.
- 1.2. General requirements for the organization of the entrance examinations. The entrance examination is conducted in person and/ or via distance learning technologies using a proctoring system.
- 1.3. Description of the format of the entrance examinations. The written examination is conducted in the form of a test and a detailed answer. The structure of the examination paper: the examination paper consists of two parts that include 27 tasks. Part 1 consists of 23 tasks in the test form that require a short answer. Part 2 consists of 4 tasks that require detailed answers.
- 1.4. The duration of the exam. The participants are given 3 hours and 55 minutes (235 minutes) to complete the examination tasks in Informatics and ICT.
- 1.5. Assessment criteria for the entrance examination. Each type of the examination task is assessed on a 100-point scale. Completed tasks in Part 1 will be given 1 point. The Part 1 task is considered completed when an examinee has given an answer that matches the correct answer code. Each task is given either 0 points ("task is not completed") or 1 point ("task is completed"). The maximum initial score that can be obtained for completing Part 1 tasks is 23 points. The Part 2 tasks are assessed from 0 to 4 points. Answers to Part 1 and Part 2 tasks will be checked and evaluated by experts. The maximum score that can be obtained for completing Part 2 tasks is 12 points. The entrance examination is considered passed if an applicant received at least 9 initial points or at least 46 final points.

Section 2. Program content

2.1. Content of the test sections. The content of the tasks has been developed according to the basic topics of the informatics and ICT course integrated in the following thematic blocks: Information and Its Coding, Modelling and Computer Experiment, Numeral Systems, Logic and Algorithms, Elements of Algorithm Theory, Programming, Architecture of Computers and Computer Networks, Processing of Numerical Information, Information Retrieval and Storage Technologies. The content of the examination paper covers the main content of the Informatics and ICT course, its core topics, the most important material in them, which is unambiguously interpreted in most versions of the Informatics and ICT course taught at school.

The examination paper contains both tasks of the basic level of difficulty testing knowledge and skills stipulated by the basic level standard and tasks of advanced and high levels



of difficulty testing knowledge and skills stipulated by the specialized level standard. On the one hand, the number of tasks in one edition of testing and assessment materials (TAM) must provide a comprehensive check of knowledge and skills of applicants that they have acquired over the entire period of study in the subject, and, on the other hand, meet the criteria of difficulty, stability of results, and reliability of measurement. For this purpose, there are two types of tasks in TAM: short-answer and detailed-answer tasks.

The structure of examination papers provides an optimal balance of different types and varieties of tasks with three levels of difficulty that test knowledge and skills over three different levels: reproducing, application in a standard situation, application in a new situation. The content of an examination paper reflects a significant part of the subject content. All this shall ensure the validity of the exam results and the reliability of the measurement.

Structure of testing and assessment materials (TAM)

Each version of an examination paper consists of two parts and includes 27 tasks that differ in form and level of difficulty.

Part 1 consists of 23 short-answer tasks.

The exam has the following types of short-answer tasks:

- tasks to calculate a certain value;
- tasks for finding the correct sequence presented as a string of symbols following a certain algorithm.

Answers to Part 1 tasks should be given in the form of a natural number or a sequence of symbols (letters or numerals) written without spaces or other separators.

Part 2 consists of 4 tasks that require detailed answers.

Part 1 contains 23 tasks of basic, advanced and high difficulty levels. This part contains short-answer tasks that require applicants to independently formulate and write down their answers in the form of a number or a sequence of symbols. The tasks test the material of all thematic blocks. In Part 1, there are 12 basic level tasks, 10 advanced level tasks, and 1 high level task.

Part 2 contains 4 tasks, the first of which is at advanced level of difficulty, the remaining 3 are high-level tasks. The tasks in this part require applicants to write a detailed answer in a free form. Part 2 tasks are aimed at testing the most important skills of writing and analyzing algorithms. These skills are tested at the advanced and high levels of difficulty. Skills related to the topic of Programming Technology are also tested at a high level of difficulty.

Distribution of tasks in TAM by content, types of skills, and methods of action.

The selection of content to be tested in TAM is based on the federal component of the state standard for secondary (complete) general education (basic and specialized levels). The distribution of tasks across the sections of the Informatics and ICT course is presented in Table 1.

Table 1

Distribution of examination paper tasks across the sections of the Informatics and ICT course

No.	Content sections	Number of tasks	Maximum initial score	Percentage of the maximum initial score for completing the tasks in this part out of the maximum initial score for the entire work equal to 35
1	Information and Its Coding	4	4	11



2	Modelling and Computer Experiment	2	2	6
3	Numeral Systems	2	2	6
4	Logic and Algorithms	6	8	23
5	Elements of Algorithm Theory	5	6	17
6	Programming	4	9	25
7	Architecture of Computers and Computer Networks	1	1	3
8	Processing of Numerical Information	1	1	3
9	Information Retrieval and Storage Technologies	2	2	6
	Total	27	35	100

TAM for Informatics and ICT does not include tasks that require simple replication of terminology, concepts, values, rules (such tasks are too easy to perform). When completing any of the TAM tasks, an examinee is required to solve a thematic problem: either directly apply a known rule, algorithm, or skill, or select the most appropriate concept or algorithm from the total number of concepts and algorithms studied and apply it in a familiar or new situation.

Knowledge of theoretical material is tested indirectly through understanding of used terminology, interrelationships between key concepts, dimensions of units, etc., when examinees complete practical tasks on various topics within the subject. Thus, the TAM in Informatics and ICT tests the understanding of theoretical material from the sections:

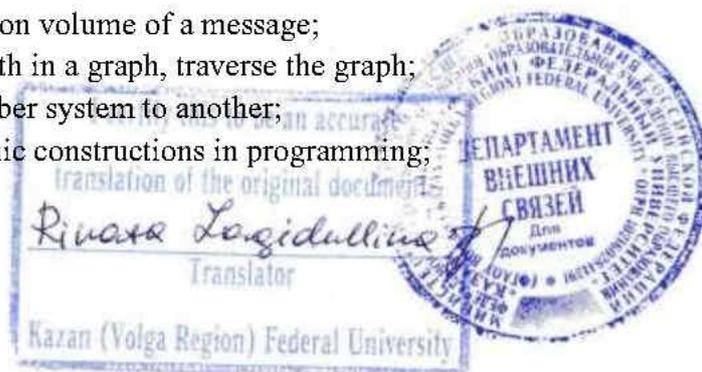
- information measurement units;
- coding principles;
- numeral systems;
- modelling
- the concept of an algorithm, its properties, and methods of recording;
- basic algorithmic constructions;
- basic concepts used in information and communication technologies.

An examination paper contains one task requiring direct application of a studied rule, formula, algorithm. This task (1) is marked as a task for reproducing knowledge and skills.

The material for testing the abilities to apply knowledge in a standard situation is included in both parts of an examination paper.

Those are the following abilities:

- to analyze the unambiguity of binary code;
- to form a truth table and logic diagram for a logical function;
- to work with data arrays;
- to calculate the information volume of a message;
- search for the shortest path in a graph, traverse the graph;
- to convert from one number system to another;
- to use standard algorithmic constructions in programming;



- to formally execute algorithms written in natural and algorithmic languages, including in programming languages;
- to determine the capacity of a computer network's address space based on the subnet mask in the TCP/IP protocol;
- formulate queries to databases and search engines.

The material for testing the abilities to apply knowledge in a new situation is also included in both parts of an examination paper. Those are the following complex abilities:

- to analyze the situation of the algorithm executor;
- to determine the base of a numeral system based on the properties of number recording;
- to describe the properties of a binary sequence according to the algorithm of its construction;
- to transform logical expressions;
- to analyze the result of an algorithm execution;
- to analyze the program text in terms of compliance of the recorded algorithm with the given task and modify it in accordance with the task;
- to build a game tree according to a given algorithm and justify a winning strategy;
- to implement a complex algorithm using modern programming systems.

Each task of an examination paper is characterized not only by tested knowledge of contented, but also by tested skills. The codifier defines two groups of requirements to the level of preparation of applicants: on the one hand, to know/understand/have a skill and, on the other hand, to use the acquired knowledge and skills in practical activities and everyday life.

Table 2 shows the distribution of tasks in terms of the skills tested in each part of examination.

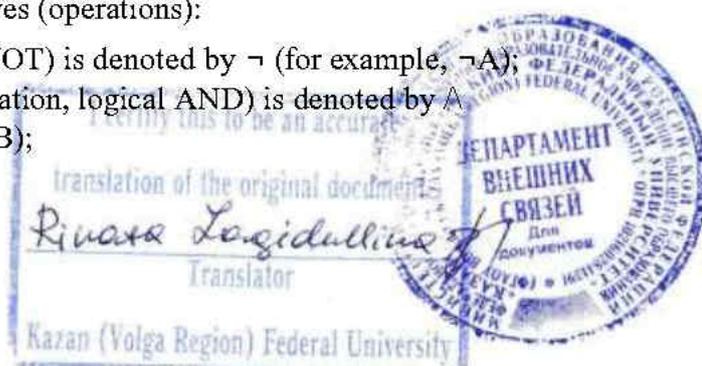
Table 2

Distribution of examination paper tasks by type of skill tested and method of action

Basic skills and methods of action	Number of tasks		
	Whole exam	Part 1	Part 2
1. Requirement to know/understand/have a skill in	24	20	4
Modelling objects, systems and processes	16	12	4
Interpretation of modelling results	4	4	0
Determination of quantitative parameters of information processes	4	4	0
2. Requirement to use the acquired knowledge and skills in practical activities and everyday life	3	3	0
Search for and select information	1	1	-
Create and use data storage structures	1	1	-
Work with common automated information systems	1	1	-
Total	27	23	4

The following conventions are used in the examination tasks.

- Symbols for logical connectives (operations):
 - negation* (inversion, logical NOT) is denoted by \neg (for example, $\neg A$);
 - conjunction* (logical multiplication, logical AND) is denoted by \wedge
 - (e.g., $A \wedge B$) or $\&$ (e.g., $A \& B$);



- d) *disjunction* (logical addition, logical OR) denoted by \vee
 - e) (e.g., $A \vee B$) or $|$ (e.g., $A | B$);
 - f) *consequence* (implication) is denoted by \rightarrow (e.g., $A \rightarrow B$);
 - g) *equality* is denoted by \equiv (for example, $A \equiv B$);
 - h) The symbol 1 is used to denote truth (a true statement); the symbol 0 is used to denote falsehood (a false statement).
2. Two logical expressions containing variables are called *equivalent* if the values of these expressions coincide at any values of the variables. Thus, expressions $A \rightarrow B$ and $(\neg A) \vee B$ are equivalent, and $A \vee B$ and $A \wedge B$ are inequivalent (the values of the expressions are different, for example, when $A = 1, B = 0$).
 3. Priorities of logical operations: inversion (negation), conjunction (logical multiplication), disjunction (logical addition), implication (consequence), equality. Thus, $\neg A \wedge B \vee C \wedge D$ means the same as $((\neg A) \wedge B) \vee (C \wedge D)$.
 4. It is possible to write $A \wedge B \wedge C$ instead of $(A \wedge B) \wedge C$. The same applies to disjunction: it is possible to write $A \vee B \vee C$ instead of $(A \vee B) \vee C$.
 5. Mbyte and Kbyte are used in the traditional meaning for Informatics: units of measurement whose ratio to the unit "byte" is expressed as a power of two.

Section 3. Assessment materials

3.1. Instruction on task completion.

Different number of points will be given for TAM tasks depending on their type.

Completed tasks in Part 1 will be given 1 point. The Part 1 task is considered completed when an examinee has given an answer that matches the correct answer code. Each task is given either 0 points ("task is not completed") or 1 point ("task is completed") (in a dichotomous grading system).

The maximum initial score that can be obtained for completing Part 1 tasks is 23 points.

The Part 2 tasks are assessed from 0 to 4 points. Answers to Part 1 and Part 2 tasks will be checked and evaluated by experts. The maximum score that can be obtained for completing Part 2 tasks is 12 points.

The distribution of tasks by parts of the examination paper is presented in Table 3.

Table 3

The distribution of tasks by parts of the examination paper

Part of examination paper	Number of tasks	Maximum initial score	Percentage of the maximum initial score for completing the tasks in this part out of the maximum initial score for the entire work equal to 35	Type of tasks
Part 1	23	23	66	short-answer tasks
Part 2	4	12	34	detailed-answer tasks
Total	27	35	100	



The answers to questions 1-23 shall be given as a a number, sequences of letters or digits, which should be written in the "answer" field after each question without spaces, commas or other additional symbols, and then click the "Next" button. Tasks 24-27 require a detailed solution. You must create a new Word document, write your answers to the questions in the document on your computer, then save it as a .pdf file and name it "ICT. Your full name". Then attach it as an attachment to the exam system. If you do not give answers for Part 2, attach a blank form and click on the "Next" button.

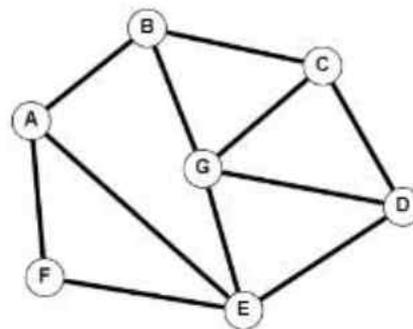
3.2. Sample tasks.

Part 1. The answers to questions 1-23 are numbers, sequences of letters or digits, which you should write in the "answer" field after each question without spaces, commas, or other additional symbols, and then click the "next" button.

1. How many natural numbers x divisible by 4 are there that satisfy inequality $650_8 \leq x \leq 3DC_{16}$? In your answer, write only the number of numbers, you do not need to write the numbers (digits) themselves.

2. In the image, the scheme of roads in the N-district is shown as a graph, and the table contains information about the length of each of these roads (in kilometers). Since the table and the scheme were drawn independently of each other, the numbering of settlements in the table does not correspond to the letters in the graph. Determine the length of the road from point B to point C. Write an integer in the answer.

	S1	S2	S3	S4	S5	S6	S7
S1		19		17		15	
S2	19		1		9		
S3		1			11		12
S4	17				17	25	5
S5		9	11	17			10
S6	15			25			
S7			12	5	10		

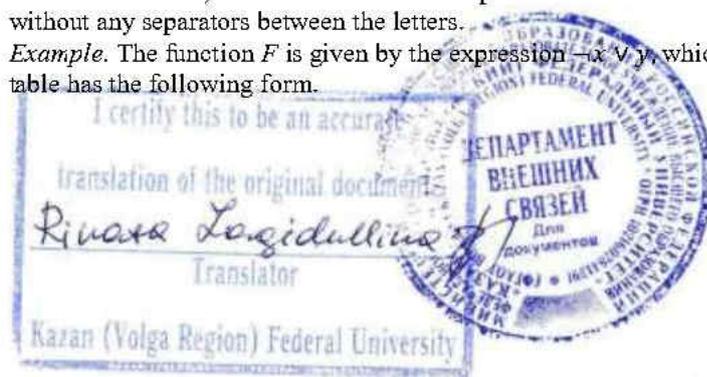


3. The function F is set by the expression $(\neg w \vee (((x \wedge y) \rightarrow \neg z) \wedge (x \vee y \vee z))) \wedge ((z \wedge w) \rightarrow x)$. The figure shows a partially filled fragment of the truth table for the function F , containing non-recurring rows. Determine to which column of the function's truth table F corresponds each of the variables x, y, z, w .

				F
		1	0	0
	1	0		0
	0	0		0
				0

In your answer, write the letters x, y, z, w in the order of the corresponding columns (first the letter that corresponds to the first column, then the letter that corresponds to the second column, etc.). Write the letters in the answer in a row without any separators between the letters.

Example. The function F is given by the expression $\neg x \vee y$, which depend on two variables, and the fragment of the table has the following form.



		<i>F</i>
0	1	0

In this case, the first column corresponds to the variable y , and the second column – corresponds to the variable x . The answer should be: yx .

4. Below there are two table fragments from the neighborhood residents' database. Each row in Table 2 contains information about a child and one of the parents. The information is represented by the value of the ID field in the corresponding row of Table 1.

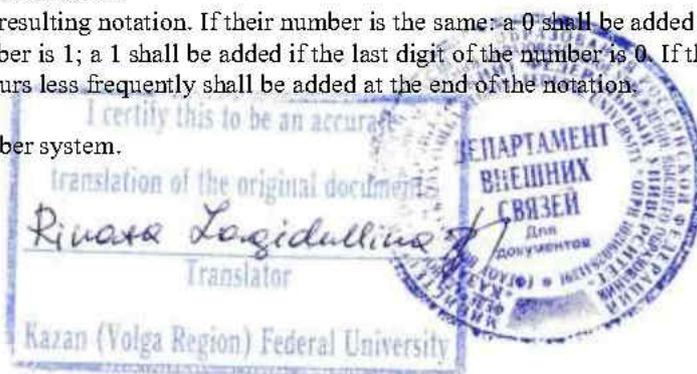
Based on the given data, determine the number of people who have two children of different sexes. When calculating your answer, only consider the information that given in the table fragments.

Table 1				Table 2	
ID	Last name, initials	Sex	Year of birth	Parent_ID	Child_ID
14	Ivanova M. A.	F	1942	24	25
24	Skanavi I. P.	M	1943	44	25
25	Skanavi P. I.	M	1973	25	26
26	Skanavi P. P.	M	1996	64	26
34	Kushchenko A. I.	F	1964	24	34
35	Kushchenko V. S.	F	1987	44	34
36	Kushchenko S. S.	M	1964	34	35
44	Lebed A. S.	F	1941	36	35
45	Lebed V. A.	M	1953	14	36
46	Gross O. S.	F	1992	34	46
47	Gross P. O.	M	2009	36	46
54	Klemenko A. P.	F	1993	25	54
64	Krot P. A.	F	1964	64	54
68	Sergeev A. P.	M	1958	46	47
...

5. Messages containing only letters from the set: E, C, O, Л, A, П, P are transmitted over the communication channel. The binary code satisfying the Fano direct condition, according to which no code word is the beginning of another code word, is used for transmission. This condition ensures that the encoded messages can be unambiguously decoded. Code words for some letters are known: C – 0, П – 111. For the five remaining letters E, O, Л, A, P the code words are unknown. What number of binary characters is required to encode the word ЛЕСОПОЛОСА if it is known that it is encoded with the minimum possible number of binary characters?

6. The algorithm receives a natural number $N > 100$ as its input. The algorithm constructs a new number R based on it as follows:

1. Construction of a binary notation of the number N.
2. Counting the number of 0 and 1 in the resulting notation. If their number is the same: a 0 shall be added at the end of the notation if the last digit of the number is 1; a 1 shall be added if the last digit of the number is 0. If the number of 0 and 1 are not equal, the digit that occurs less frequently shall be added at the end of the notation.
3. Step 2 is repeated one more time.
4. The result is converted to decimal number system.



What is the smallest initial number N that, when processed by the algorithm, will result in a number that, when divided by 4, gives a remainder of 3?

7. The executor Turtle operates on a plane with Cartesian coordinate system. At the initial moment, the Turtle is at the point of origin, its head is pointing along the positive direction of the ordinate axis, and its tail is down. With its tail down, the Turtle leaves a line-shaped mark on the field. At any specific moment, we know the position of the executor and the direction of its movement. There are two commands for the executor:

Forward n (where n is an integer) causing the Turtle to move n units in the direction its head is pointing, and

Right m (where m is an integer) causing the direction to change by m degrees clockwise.

Repeat k [Command1 Command2 ... CommandS] means that the sequence of S commands will be repeated k times.

The Turtle was given the following algorithm:

Repeat 31 [Forward 9 Right 60 Right 120 Forward 9 Right 36].

Determine how many line segments the figure generated by this algorithm will consist of.

8. A musical piece was recorded in mono, digitized and saved as a file without using data compression. The size of the resulting file is 112 Mbyte. The same musical piece was then rerecorded in stereo (dual-channel recording) and digitized at resolution 3.5 times higher and sampling rate 2 times lower than the first time. No data compression was done. Specify the size of the resulting rerecorded file in Mbyte. Write only an integer in your answer; do not write the unit of measurement.

9. Determine how many five-digit numbers, not divisible by 3, written in the senary numeral system (base-6), have only one digit "2" with no even and odd digits next to each other.

10. Below is a fragment of a spreadsheet. A formula was copied from cell C2 to cell D3. When copying, cell addresses in the formula changed automatically. What is the numerical value of the formula in cell D3? In your answer, write down only the number.

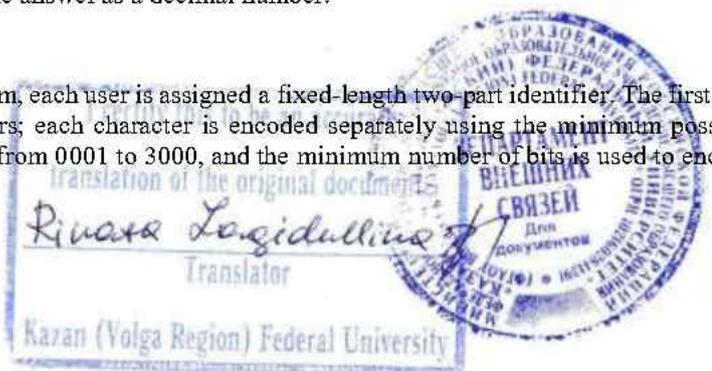
	A	B	C	D	E
1	6	15	830	700	4500
2	55	88	=MAX(A\$1:A\$4)	2500	4200
3	46	15	925	784	7500
4	15	70	50	690	555

Note. The \$ sign means absolute addressing.

11. In TCP/IP terminology, the network mask is a binary number that determines which part of the IP address of a node refers to the network address and which part refers to the address of the node itself on that network. Typically, the mask is written using the same rules as the IP address: in the form of 4 bytes with each byte written as a decimal number. In this case, the mask firstly (in the high-order digits) contains 1, and then from a certain order - 0. The network address is obtained by applying a bitwise conjunction to a given IP-address of the node and to the mask. For example, if the IP address of the node is 231.32.255.131, and the mask is 255.255.240.0, then the network address is 231.32.240.0.

For the node with the IP address 55.102.221.201 the network address is 55.102.221.192. What is the highest value of the last (rightmost) byte of the mask? Write the answer as a decimal number.

12. When registering in a computer system, each user is assigned a fixed-length two-part identifier. The first part includes 19 capital and uppercase Latin letters; each character is encoded separately using the minimum possible number of bits. The second part is an integer from 0001 to 3000, and the minimum number of bits is used to encode



it. An integer number of bytes is allocated for encoding the entire identifier. In addition, additional information is stored for each user (also integer number of bytes, the same for each user). Determine how many bytes are required for additional information for one user if the data for 72 users occupies 4536 bytes. *Note:* Latin alphabet has 26 letters.

13. The executor Editor receives a string of digits as an input and converts it. The Editor can execute two commands, in both commands v and w represent string of digits.

A) **replace** (v, w).

This command replaces the first leftmost occurrence of the string v in the line with the string w . For example, execution of the command **replace** (111, 27) converts string 05111150 to string 0527150. If the line does not contain any occurrences of the string v , then execution of the **replace** (v, w) command does not change this string.

B) **found** (v).

This command checks if the string v occurs in the Editor's line. If it occurs, the command returns "true", otherwise the command returns "false". The executor's row will not be changed in this case.

The loop

```
WHILE condition
    command sequence
END WHILE
```

is executed as long as the condition is true.

In construction

```
IF condition
    THEN command1
    ELSE command2
END IF
```

command1 (if the *condition* is true) or *command2* (if the *condition* is false) is executed.

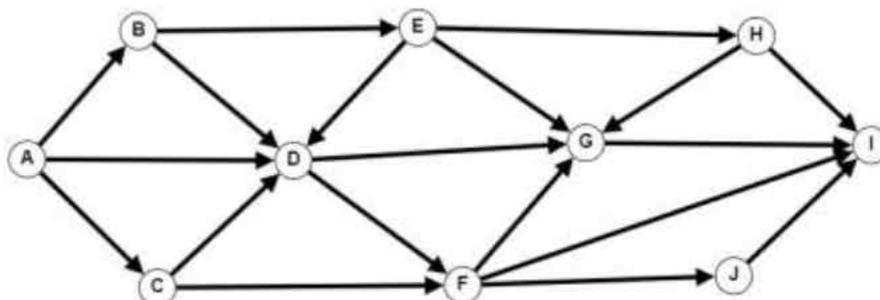
It is known that the initial string starts with digit "0" and contains 25 digits "1", 30 digits "2", and n digits "3" placed in random order. At what minimum value of n will the sum of digits in the line resulting from the algorithm be divisible by 100?

START

```
    WHILE found (01) OR found (02) OR found (03)
        replace (01, 1103)
        replace (02, 201)
        replace (03, 20)
    END WHILE
```

END

14. The figure shows a road map connecting cities A, B, C, D, E, F, G, H, I, J. It is possible to move along each road in only one direction, indicated by an arrow. How many different paths are there from city A to city I that pass through city G?



15. The value of the arithmetic expression



$$6 * 49^{30} + 42 * 7^{22} + 27 * 7^{14} - 13$$

is written in 7-base number system. How many 6 are contained in this notation?

16. Three line segments are given on the number line: $A = [5; 42]$, $B = [77; 120]$ and $C = [21; 97]$. Specify how many integer values of x , for which the Boolean expression

$$(\neg(x \in A) \rightarrow (x \in B)) \vee ((x \in C) \rightarrow (x \in B)) \vee \neg \text{ДЕЛ}(x, 7)$$

is false (i.e., takes on value 0). Logical statement $\text{ДЕЛ}(n, m)$ means "natural number n is divisible by natural number m leaving no remainder".

17. A recursive algorithm F is written below in three languages. Write down all the numbers that will be displayed on the screen when the call $F(6)$ is executed, without spaces or separators. The numbers should be written in the same order in which they will be displayed on the screen.

Pascal	Python	C++
<pre> procedure F(n: integer); begin if n > 0 then if (n mod 2 <> 0) then begin F(n + 1); writeln(n + 3) end else begin writeln (n + 2); F(n - 3) end; end; </pre>	<pre> def F(n): if n > 0: if (n % 2 != 0): F(n + 1) print(n + 3) else: print(n + 2) F(n - 3) </pre>	<pre> void F(int n) { if (n > 0) if (n % 2 != 0) { F(n + 1); std::cout << n + 3 << std::endl; } else { std::cout << n + 2 << std::endl; F(n - 3); } } </pre>

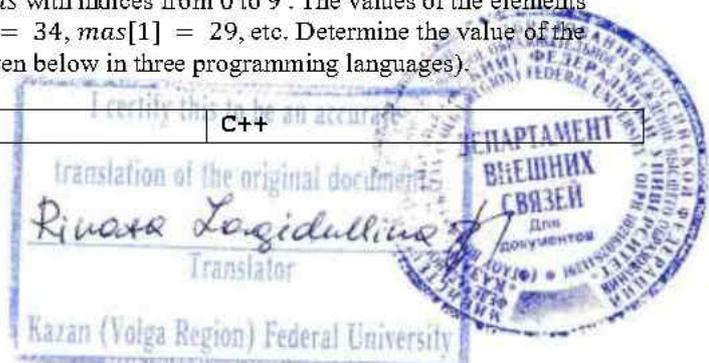
18. In the query language of the search server, the symbol "|" is used to denote the logical operation "OR" and the symbol "&" is used to denote the logical operation "AND". The table shows the queries and the number of pages of some Internet segment found upon them.

Query	Pages found (in hundreds of thousands)
<i>Logic & Informatics</i>	150
<i>Programming</i>	750
<i>Logic</i>	375
<i>Informatics & Programming</i>	230
<i>Informatics Programming Logic</i>	1265
<i>Logic & Programming</i>	0

How many pages (in hundreds of thousands) will be found for the query "Informatics"?

19. The program uses a one-dimensional integer array mas with indices from 0 to 9. The values of the elements are 34, 29, 28, 3, 9, 6, 40, 7, 50, 4 respectively, i.e. $mas[0] = 34$, $mas[1] = 29$, etc. Determine the value of the variable s after executing the following program fragment (given below in three programming languages).

Pascal	Python	C++
--------	--------	-----



<pre>s := 0; n := 9; for i := 0 to 8 do if mas[i] >= mas[i+1] then begin t := mas[i + 1]; mas[i + 1] = mas[i]; mas[i] := mas[n]; mas[n] := t; s += mas[i] end;</pre>	<pre>s = 0 n = 9 for i in range(9): if mas[i] >= mas[i + 1]: t = mas[i + 1] mas[i + 1] = mas[i] mas[i] = mas[n] mas[n] = t s += mas[i]</pre>	<pre>s = 0, n = 9; for (int i=0; i<9; i++){ if (mas[i] >= mas[i+1]){ int t = mas[i + 1]; mas[i + 1] = mas[i]; mas[i] = mas[n]; mas[n] = t; s += mas[i]; }</pre>
---	---	---

20. The algorithm is written below in three programming languages. Give the smallest value of a greater than 100, upon entering which the algorithm will display 17.

Pascal	Python	C++
<pre>var a, b, t :integer; begin readln(a); b := a - 34; while (a <> b) do begin a := a - b; if b > a then begin t := a; a := b; b := t; end end; writeln(a); end.</pre>	<pre>a = int(input()) b = a - 34 while a != b: a -= b if b > a: t = a a = b b = t print(a)</pre>	<pre>int a, b, t; std::cin>>a; b = a - 34; while (a != b){ a -= b; if (b > a){ t = a; a = b; b = t; } } std::cout<<a<<std::endl;</pre>

21. The algorithm is written below in three programming languages. Write the number that will be displayed as a result of its execution.

Pascal	Python	C++
<pre>var a,b,M,R,t:integer; function F(x:integer):integer; begin F:= -2*(x*x-4)*(x*x-4)+5; end; begin a := -3; b := 3; M := a; R := F(a); for t:=a to b do if F(t) <= R then begin M := t; R :=F(t) end; writeln(R+200); end.</pre>	<pre>def F(x): return -2*(x*x-4)*(x*x-4)+5 a = -3 b = 3 M = a R = F(a) for t in range(a, b + 1): if F(t) <= R: M = t R =F(t) print(R+200)</pre>	<pre>#include <iostream> using namespace std; long F(long x) { return -2*(x*x-4)*(x*x-4)+5; } int main() { long a, b, t, M, R; a = -3; b = 3; M = a; R = F(a); for (t = a; t <= b; t++) { if (F(t) <= R) { M = t; R = F(t); } } cout << R+200<< endl; return 0; }</pre>



22. The table provides information about a set of N computational processes that can be executed in parallel or sequentially. We shall say that process B depends on process A if the results of execution of process A are necessary for the execution of process B. In this case processes can be executed only sequentially. The first column of the table contains the process identifier (ID), the second column of the table contains its execution time in milliseconds, and the third column lists the IDs of the processes (separated by ";") on which this process depends. If the process is independent, the table indicates 0.

Process B ID	Process B execution time (msec)	Process(es) A ID
1	4	0
2	4	0
3	2	0
4	8	2
5	10	1;2;3
6	4	0
7	7	4;6
8	6	5;7
9	6	3;5;8
10	8	8;9;

Determine the **minimum** time after which the execution of the entire set of processes will be completed, provided that all independent processes can be executed in parallel.

23. The Executor converts the number on the screen. The Executor has three commands that have been assigned the following numbers:

1. **Add 2**
2. **Add 3**
3. **Multiply by 5**

A program for the Executor – is a series of commands. How many programs are there for which at initial number 1 the result is the number 25, and at the same time the computation trajectory contains the number 8 and does not contain the number 20?

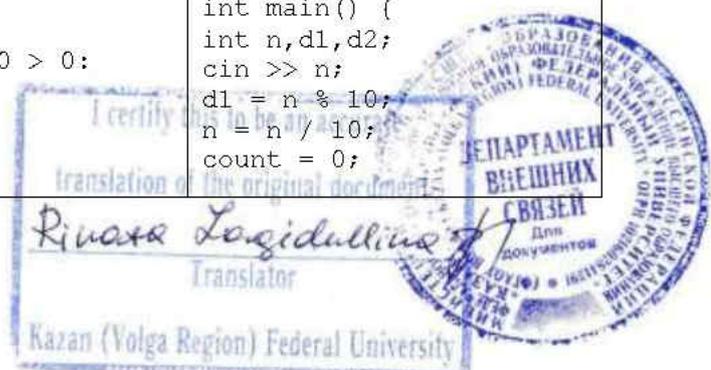
The program computation trajectory – is the sequence of results of execution of all program commands. For example, for program 123 with the initial number 5, the trajectory will consist of numbers 7, 10, 50.

Part 2

Use the ANSWER SHEET No. 2 to record your answers to this part (25-27). First, write the number of the task (25, 26 etc.) then write the complete solution. Answers should be written down clearly and legibly.

24. A natural number greater than 100 and not exceeding 10^9 , is entered for processing. Write a program that displays the number of digits of a number that are greater than or equal to the previous digit in the number (if the previous digit exists). The programmer wrote the program incorrectly. Below this program is given in three programming languages for your convenience.

Pascal	Python	C++
<pre>var n,d1,d2: longint; begin readln(n); d1 := n mod 10; n := n div 10; count := 0; while n div 10 > 0 then begin d2 := n mod 10;</pre>	<pre>n = int(input()) d1 = n % 10 n //= 10 count = 0 while n // 10 > 0: d2 = n % 10 if d2 <= d1: count += 1 d2 = d1</pre>	<pre>#include <iostream> using namespace std; int main() { int n,d1,d2; cin >> n; d1 = n % 10; n = n / 10; count = 0;</pre>



<pre>if d2 <= d1 then count += 1; d2 := d1; n = n div 10; end; writeln(p) end.</pre>	<pre>n //= 10 print(count)</pre>	<pre>while (n / 10 > 0) { d2 = n % 10; if (d2 <= d1) count += 1; d2 = d1; n = n / 10; } cout << count << endl; return 0; }</pre>
---	----------------------------------	--

- Write what this program will display when you enter the number 123.
- Give the maximum three-digit number upon entry of which the program will display the correct answer despite errors.
- Find the errors made by the programmer and correct them. The error correction should involve only the line in which the error occurs. For each error:
 - write out the line in which the error was made;
 - specify how to correct the error, i.e. give the correct version of the line.

It is known that you can correct exactly two lines in the program text so that it will work correctly.

It is enough to specify errors and the way to correct them for one programming language. Please note that you are required to find and correct exactly two errors in the existing program, rather than writing your own program, which may use a different solution algorithm.

25. An integer array of 50 elements is given. The array elements are numbered from 0 to 49 and can take natural values from 1 to 10000 inclusive. Using one of the programming languages, describe an algorithm that finds the number of pairs of array elements in which both numbers in the pair are divisible by 2 and exactly one ends in 6, and then replaces each element that is not divisible by 6 with the value found. As a result, a modified array should be output, with each element written on a new line. A pair refers to two adjacent array elements.

For example, for an initial array of six elements:

2
6
8
11
36
15

the program should output the following array:

2
6
2
2
36
2

Pascal	Python	C++
<pre>const N = 50; var a: array [0..N-1] of longint; i, j, k: longint; begin for i := 0 to N-1 do readln(a[i]); ... end.</pre>	<pre># it is also allowed # to use two # integer variables j and k a = [] n = 50 for i in range(n): a.append(int(input())) ...</pre>	<pre>#include <iostream> using namespace std; const int N = 50; int main(){ long a[N]; long i, j, k; for (i = 0; i < N; i++) cin >> a[i]; ... return 0; }</pre>



As an answer, you need to provide a fragment of the program that should be in place of the ellipsis. You can also write the solution in another programming language (specify the name and version of the programming language used, e.g. Free Pascal 2.6). In this case, you must use the same original data and variables as given in the specification.

26. Two players, Petya and Vanya, play the following game. Two piles of stones lie in front of the players. Players take turns, the first move is made by Petya. In one move a player can either add three stones to one of the piles (at his choice) or multiply by four the number of stones in a pile. For example, let there be given 10 stones in one pile and 5 stones in the other; we will denote this position in the game as (10, 5). Then, in one move, you can get any of the four positions: (13, 5), (40, 5), (10, 8), (10, 20). To make moves, each player has an unlimited number of stones.

The game ends when the total number of stones in the piles becomes at least 105. The winner is the player who made the last move, i.e., the first to reach a position where the total number of stones in the piles is 105 or more. At the starting moment there were 2 stones in the first pile, and S stones in the second pile, $1 < S < 103$.

We shall say that a player has a winning strategy if he can win regardless of his opponent's moves. Describing a player's strategy – means describing what move they should make in any situation they may encounter when the opponent makes different moves. The description of a winning strategy should not include moves made by a player using this strategy that are not unconditionally winning for him, i.e., that are not the winning ones regardless of his opponent's moves. Complete the following tasks. In all cases, justify your answer.

1) Specify the value of S , at which Petya cannot win in one move, but at any move of Petya, Vanya can win with his first move.

2) Find the minimum and maximum values of S at which Petya has a winning strategy, and two conditions are simultaneously satisfied:

- Petya can win in one move;
- Petya can win with his second move regardless of how Vanya moves.

Describe Petya's winning strategy.

3) Find the value of S at which two conditions are simultaneously satisfied:

- Vanya has a winning strategy that allows him to win with the first or the second move regardless of how Petya plays;
- Vanya does not have a strategy that will allow him to win with his first move.

If several values of S are found, write the maximum value in your answer. For the found value of S , describe Vanya's winning strategy. Construct a tree of all possible moves for Vanya's winning strategy (in the form of a figure or table). Indicate moves on the edges of the tree and positions on its nodes.

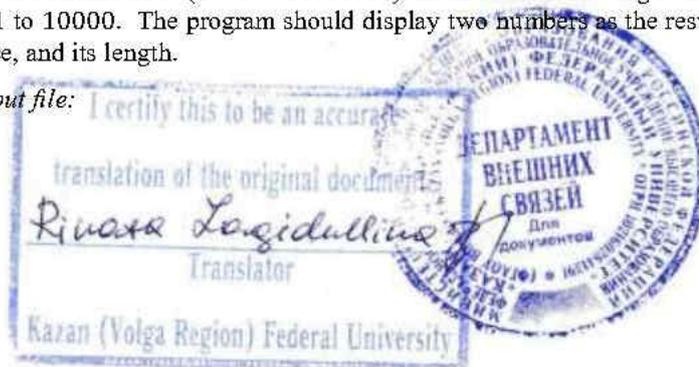
27. A sequence of N natural numbers is given. Numbers in the sequence can take values from 1 to 10000. All its continuous subsequences are considered such that the sum of the elements of each of them is divisible by 13 and the product is divisible by 6. Find the subsequence with the maximum sum and its length. If you find several such subsequences, write the number of elements in the longest of them in your answer.

Description of input and output data

The first line of the input file specifies the number of numbers N ($1 < N < 10000000$). Each of the following N lines contains one natural number in the range from 1 to 10000. The program should display two numbers as the result: the maximum sum of the continuous subsequence, and its length.

An example of organizing original data in an input file:

10
1
97
68
11
46
96
7
29



41
12

Example of input data for the input data example above:
325 6

Explanation: The sum equal to 325, is the result of summing the elements 97, 68, 11, 46, 96, 7... Among these elements, there is a number that is divisible by 6, which is 96.

You need to write a time-efficient program to solve the described problem. A program is considered to be time-efficient if, when the number of sequence elements N increases by k times, the time of program operation increases by no more than k times. The maximum score for a correct time-efficient program (one that contains no syntax errors and gives the correct answer for any acceptable input data) is 4 points. Maximum score for a correct program that does not meet the time-efficiency requirements is – 2 points.

You may write one or two problem solving programs (e.g., one of the programs may be less effective). If you write two programs, each program will be graded independently of the other, and the higher of the two grades will be the final grade. Before the program text, be sure to briefly describe the solution algorithm. Specify the programming language that you use and its version.

Check that each answer is written next to the corresponding task number.

Section 4. The list of literature and information sources to prepare for entrance examination

- Information and educational portal. Documents, materials, manuals, sample tasks to the Unified State Exam (EGE), State Final Examination (GIA). [Electronic resource]. - Access mode: <http://egeigia.ru/>;
- Official website of the Federal Education and Science Supervision Service (Rosobrnadzor) “Unified State Exam (EGE) Portal. We know everything about EGE”. [Electronic resource]. – Access mode: <http://4ege.ru/>;
- Methodological materials and software on Informatics - Access mode: <http://kpolyakov.spb.ru>
- Ugrinovich N. D. Informatics and ICT. Specialized Level: Textbook for 10th Grade – BINOM. Knowledge Laboratory, 2012
- Ugrinovich N. D. Informatics and ICT. Specialized Level: Textbook for 11th Grade – BINOM. Knowledge Laboratory, 2012
- V. G. Davydov. Programming and Basics of Algorithmization - Moscow: Higher School, 2003

