

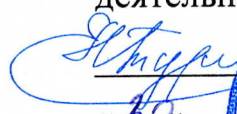
МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
«Казанский (Приволжский) федеральный университет»

Институт информационных технологий и интеллектуальных систем

УТВЕРЖДАЮ

Проректор по образовательной

деятельности


Е.А. Турилова
« 30 » 2023 г.



**ПРОГРАММА ВСТУПИТЕЛЬНОГО ИСПЫТАНИЯ
ПО ИНФОРМАТИКЕ И ИКТ**

Лист согласования программы вступительного испытания

Разработчик программы:

Доцент кафедры Программной инженерии

Хадиев К.Р.

Председатель экзаменационной комиссии

М.М. Абрамский

Программа вступительного испытания обсуждена и одобрена на заседании кафедры Программной инженерии Института информационных технологий и интеллектуальных систем Протокол №1 от «29» сентября 2023 г.

Решением Учебно-методической комиссии Института информационных технологий и интеллектуальных систем Программа вступительного испытания рекомендована к утверждению Ученым советом, Протокол № 1 от «29» сентября 2023 г.

Программа вступительного испытания утверждена на заседании Ученого совета Института информационных технологий и интеллектуальных систем, Протокол №1 от «29» сентября 2023 г.

Содержание

Раздел I. Вводная часть

- 1.1 Цель и задачи вступительных испытаний
- 1.2 Общие требования к организации вступительных испытаний
- 1.3 Описание формы проведения вступительных испытаний
- 1.4 Продолжительность вступительных испытаний
- 1.5 Структура вступительных испытаний

Раздел II. Содержание программы

- 2.1 Содержание разделов тестирования

Раздел III. Фонд оценочных средств

- 3.1. Инструкция по выполнению работы
- 3.2. Примерные задания

Раздел IV. Список литературы

Раздел 1. Вводная часть

- 1.1. Цели и задачи вступительных испытаний;
- 1.2. Общие требования к организации вступительных испытаний. Форма проведения вступительного испытания очно и/или с использованием дистанционных технологий с использованием системы прокторинга.
- 1.3. Описание формы проведения вступительных испытаний. Письменный экзамен в виде тестирования и развернутого ответа. Структуру экзаменационного билета: экзаменационная работа состоит из двух частей, включающих в себя 27 заданий. Часть 1 содержит 23 задания с кратким ответом в виде теста. Часть 2 содержит 4 задания с развернутым ответом.
- 1.4. Продолжительность экзамена. На выполнение экзаменационной работы по информатике и ИКТ отводится 3 часа 55 минут (235 минут).
- 1.5. Критерии оценки выполнения каждого вида экзаменационного задания по 100-балльной шкале в случае успешного и неудовлетворительного прохождения вступительных испытаний. Баллы, полученные Вами за выполненные задания, суммируются.

В соответствии с Порядком проведения государственной итоговой аттестации по образовательным программам среднего общего образования (приказ Минобрнауки России от 26.12.2013 № 1400 зарегистрирован Минюстом России 03.02.2014 № 31205) «61. По результатам первой и второй проверок эксперты независимо друг от друга выставляют баллы за каждый ответ на задания экзаменационной работы ЕГЭ с развернутым ответом. 62. В случае существенного расхождения в баллах, выставленных двумя экспертами, назначается третья проверка. Существенное расхождение в баллах определено в критериях оценивания по соответствующему учебному предмету. Эксперту, осуществляющему третью проверку, предоставляется информация о баллах, выставленных экспертами, ранее проверявшими экзаменационную работу». Если расхождение составляет 2 и более балла за выполнение любого из заданий 24–27, то третий эксперт проверяет ответы только на те задания, которые вызвали столь существенное расхождение.

Постарайтесь выполнить как можно больше заданий и набрать наибольшее количество баллов.

Раздел 2 «Содержание программы»

2.1 Содержание заданий разработано по основным темам курса информатики и ИКТ, объединенных в следующие тематические блоки: «Информация и ее кодирование», «Моделирование и компьютерный эксперимент», «Системы счисления», «Логика и алгоритмы», «Элементы теории алгоритмов», «Программирование», «Архитектура компьютеров и компьютерных сетей», «Обработка числовой информации», «Технологии поиска и хранения информации». Содержанием экзаменационной работы охватывается основное содержание курса информатики и ИКТ, важнейшие его темы, наиболее значимый в них материал, однозначно трактуемый в большинстве преподаваемых в школе вариантов курса информатики и ИКТ.

Работа содержит как задания базового уровня сложности, проверяющие знания и умения, предусмотренные стандартом базового уровня, так и задания повышенного и

высокого уровней сложности, проверяющие знания и умения, предусмотренные стандартом профильного уровня. Количество заданий в варианте КИМ должно, с одной стороны, обеспечить всестороннюю проверку знаний и умений выпускников, приобретенных за весь период обучения по предмету, и, с другой стороны, соответствовать критериям сложности, устойчивости результатов, надежности измерения. С этой целью в КИМ используются задания двух типов: с кратким ответом и развернутым ответом.

Структура экзаменационной работы обеспечивает оптимальный баланс заданий разных типов и разновидностей, трех уровней сложности, проверяющих знания и умения на трех различных уровнях: воспроизведения, применения в стандартной ситуации, применения в новой ситуации. Содержание экзаменационной работы отражает значительную часть содержания предмета. Все это обеспечивает валидность результатов экзамена и надежность измерения.

Структура КИМ

Каждый вариант экзаменационной работы состоит из двух частей и включает в себя 27 заданий, различающихся формой и уровнем сложности.

Часть 1 содержит 23 задания с кратким ответом.

В экзаменационной работе предложены следующие разновидности заданий с кратким ответом:

- задания на вычисление определенной величины;
- задания на установление правильной последовательности, представленной в виде строки символов по определенному алгоритму.

Ответ на задания части 1 дается соответствующей записью в виде натурального числа или последовательности символов (букв или цифр), записанных без пробелов и других разделителей.

Часть 2 содержит 4 задания с развернутым ответом.

Часть 1 содержит 23 задания базового, повышенного и высокого уровней сложности. В этой части собраны задания с кратким ответом, подразумевающие самостоятельное формулирование и запись ответа в виде числа или последовательности символов. Задания проверяют материал всех тематических блоков. В части 1 12 заданий относятся к базовому уровню, 10 заданий – к повышенному уровню сложности, 1 задание – к высокому уровню сложности.

Часть 2 содержит 4 задания, первое из которых повышенного уровня сложности, остальные 3 задания высокого уровня сложности. Задания этой части подразумевают запись развернутого ответа в произвольной форме. Задания части 2 направлены на проверку сформированности важнейших умений записи и анализа алгоритмов. Эти умения проверяются на повышенном и высоком уровнях сложности. Также на высоком уровне сложности проверяются умения по теме «Технология программирования».

Распределение заданий КИМ по содержанию, видам умений и способам действий.

Отбор содержания, подлежащего проверке в КИМ, осуществляется на основе Федерального компонента государственного стандарта среднего (полного) общего образования (базовый и профильный уровни). Распределение заданий по разделам курса информатики и ИКТ представлено в таблице 1.

*Распределение заданий экзаменационной работы
по содержательным разделам курса информатики и ИКТ*

№	Содержательные разделы	Количество заданий	Максимальный первичный балл	Процент максимального первичного балла за выполнение заданий данного раздела от максимального первичного балла за всю работу, равного 35
1	Информация и ее кодирование	4	4	11
2	Моделирование и компьютерный эксперимент	2	2	6
3	Системы счисления	2	2	6
4	Логика и алгоритмы	6	8	23
5	Элементы теории алгоритмов	5	6	17
6	Программирование	4	9	25
7	Архитектура компьютеров и компьютерных сетей	1	1	3
8	Обработка числовой информации	1	1	3
9	Технологии поиска и хранения информации	2	2	6
	Итого	27	35	100

Таблица 1.

В КИМ по информатике и ИКТ не включены задания, требующие простого воспроизведения знания терминов, понятий, величин, правил (такие задания слишком просты для выполнения). При выполнении любого из заданий КИМ от экзаменуемого требуется решить тематическую задачу: либо прямо использовать известное правило, алгоритм, умение, либо выбрать из общего количества изученных понятий и алгоритмов наиболее подходящее и применить его в известной или новой ситуации.

Знание теоретического материала проверяется косвенно через понимание используемой терминологии, взаимосвязей основных понятий, размерностей единиц и т.д. при выполнении экзаменуемыми практических заданий по различным темам предмета. Таким образом, в КИМ по информатике и ИКТ проверяется освоение теоретического материала из разделов:

- единицы измерения информации;
- принципы кодирования;
- системы счисления;
- моделирование;
- понятие алгоритма, его свойств, способов записи;
- основные алгоритмические конструкции;
- основные понятия, используемые в информационных и коммуникационных технологиях.

Экзаменационная работа содержит одно задание, требующее прямо применить изученное правило, формулу, алгоритм. Это задание (1) отмечено как задание на воспроизведение знаний и умений.

Материал на проверку сформированности умений применять свои знания в стандартной ситуации входит в обе части экзаменационной работы.

Это следующие умения:

- анализировать однозначность двоичного кода;
- формировать для логической функции таблицу истинности и логическую схему;

- оперировать массивами данных;
- подсчитать информационный объем сообщения;
- искать кратчайший путь в графе, осуществлять обход графа;
- осуществлять перевод из одной системы счисления в другую;
- использовать стандартные алгоритмические конструкции при программировании;
- формально исполнять алгоритмы, записанные на естественных и алгоритмических языках, в том числе на языках программирования;
- определять мощность адресного пространства компьютерной сети по маске подсети в протоколе TCP/IP;
- оценить результат работы известного программного обеспечения;
- формулировать запросы к базам данных и поисковым системам.

Материал на проверку сформированности умений применять свои знания в новой ситуации также входит в обе части экзаменационной работы. Это следующие сложные умения:

- анализировать обстановку исполнителя алгоритма;
- определять основание системы счисления по свойствам записи чисел;
- описывать свойства двоичной последовательности по алгоритму ее построения;
- осуществлять преобразования логических выражений;
- моделировать результаты поиска в сети Интернет;
- анализировать результат исполнения алгоритма;
- анализировать текст программы с точки зрения соответствия записанного алгоритма поставленной задаче и изменять его в соответствии с заданием;
- умение построить дерево игры по заданному алгоритму и обосновать выигрышную стратегию;
- реализовывать сложный алгоритм с использованием современных систем программирования.

Каждое задание экзаменационной работы характеризуется не только проверяемым содержанием, но и проверяемыми умениями. Кодификатор определяет две группы требований к уровню подготовки выпускников: с одной стороны, знать/понимать/уметь и, с другой стороны, использовать приобретенные знания и умения в практической деятельности и повседневной жизни.

При том что стандарт образования по информатике и ИКТ содержит достаточно много требований к использованию приобретенных знаний и умений в практической жизни, используемая стандартизированная бланковая технология единого государственного экзамена не позволяет проверить выполнение этих требований в полном объеме. В работе всего 3 таких задания, они расположены в части 1 работы. Их выполнение дает менее 10% первичных баллов. Остальные 90% первичных баллов экзаменуемый может получить за счет реализации умений оперировать с теоретическим материалом предмета информатики и ИКТ. В таблице 2 характеризуется распределение заданий с точки зрения проверяемых умений в каждой части работы.

*Распределение заданий экзаменационной работы
по видам проверяемых умений и способам действий*

Основные умения и способы действий	Количество заданий		
	Вся работа	Часть 1	Часть 2
1. Требования: «Знать/понимать/уметь»	24	20	4
Моделирование объектов, систем и процессов	16	12	4
Интерпретация результатов моделирования	4	4	0
Определение количественных параметров информационных процессов	4	4	0
2. Требования: «Использовать приобретенные знания и умения в практической деятельности и повседневной жизни»	3	3	0
Осуществлять поиск и отбор информации	1	1	–
Создавать и использовать структуры хранения данных	1	1	–
Работать с распространенными автоматизированными информационными системами	1	1	–
Итого	27	23	4

Таблица 2.

В экзаменационных заданиях используются следующие соглашения.

- Обозначения для логических связей (операций):
 - отрицание* (инверсия, логическое НЕ) обозначается $\neg$ (например, $\neg A$);
 - конъюнкция* (логическое умножение, логическое И) обозначается $\wedge$
 - (например, $A \wedge B$) либо $\&$ (например, $A \& B$);
 - дизъюнкция* (логическое сложение, логическое ИЛИ) обозначается $\vee$
 - (например, $A \vee B$) либо $|$ (например, $A | B$);
 - следование* (импликация) обозначается $\rightarrow$ (например, $A \rightarrow B$);
 - тождество* обозначается $\equiv$ (например, $A \equiv B$). Выражение $A \equiv B$ истинно тогда и только тогда, когда значения A и B совпадают (либо они оба истинны, либо они оба ложны);
 - символ 1 используется для обозначения истины (истинного высказывания); символ 0 – для обозначения лжи (ложного высказывания).
- Два логических выражения, содержащих переменные, называются *равносильными* (эквивалентными), если значения этих выражений совпадают при любых значениях переменных. Так, выражения $A \rightarrow B$ и $(\neg A) \vee B$ равносильны, а $A \vee B$ и $A \wedge B$ неравносильны (значения выражений разные, например, при $A = 1, B = 0$).
- Приоритеты логических операций: инверсия (отрицание), конъюнкция (логическое умножение), дизъюнкция (логическое сложение), импликация (следование), тождество. Таким образом, $\neg A \wedge B \vee C \wedge D$ означает то же, что и $((\neg A) \wedge B) \vee (C \wedge D)$.
- Возможна запись $A \wedge B \wedge C$ вместо $(A \wedge B) \wedge C$. То же относится и к дизъюнкции: возможна запись $A \vee B \vee C$ вместо $(A \vee B) \vee C$.

5. Обозначения Мбайт и Кбайт используются в традиционном для информатики смысле – как обозначения единиц измерения, чьё соотношение с единицей «байт» выражается степенью двойки.

Раздел 3 «Фонд оценочных средств»

3.1 Инструкция по выполнению работы.

Задания КИМ оцениваются разным количеством баллов в зависимости от их типа.

Выполнение каждого задания части 1 оценивается в 1 балл. Задание части 1 считается выполненным, если экзаменуемый дал ответ, соответствующий коду верного ответа. За выполнение каждого задания присваивается (в дихотомической системе оценивания) либо 0 баллов («задание не выполнено»), либо 1 балл («задание выполнено»).

Максимальное количество первичных баллов, которое можно получить за выполнение заданий части 1, – 23.

Выполнение заданий части 2 оценивается от 0 до 4 баллов. Ответы на задания части 1 и 2 проверяются и оцениваются экспертами. Максимальное количество баллов, которое можно получить за выполнение заданий части 2, – 12.

Распределение заданий по частям экзаменационной работы представлено в таблице 3.

Распределение заданий по частям экзаменационной работы

Часть работы	Количество заданий	Максимальный первичный балл	Процент максимального первичного балла за выполнение заданий данной части от максимального первичного балла за всю работу, равного 35	Тип заданий
Часть 1	23	23	66	С кратким ответом
Часть 2	4	12	34	С развернутым ответом
Итого	27	35	100	

Таблица 3

Ответы к заданиям 1–23 записываются в виде числа, последовательности букв или цифр, которые следует записать в поле «ответа» после каждого вопроса, без пробелов, запятых и других дополнительных символов и нажать кнопку «далее». Задания 24–27 требуют развернутого решения. Вы должны создать новый документ в редакторе Word, далее пишете ответы на вопросы в этом документе на компьютере, затем сохраняете в формате .pdf и называете «ИКТ. Ваше ФИО», далее прикрепляете как вложенный файл в систему для проведения экзамена. Если вы не будете отвечать на Часть 2, то вы прикрепляете пустой бланк и нажимаете на кнопку «далее».

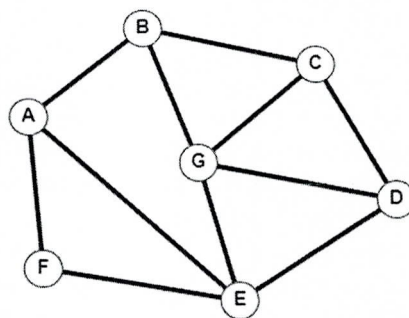
3.2. Примерные задания.

Часть 1. Ответами к заданиям 1–23 являются число, последовательность букв или цифр, которые следует записать в поле «ответа» после каждого вопроса, без пробелов, запятых и других дополнительных символов и нажать кнопку «далее».

- Сколько существует натуральных чисел x , кратных 4, для которых выполняется неравенство $650_8 \leq x \leq 3DC_{16}$? В ответе укажите только количество чисел, сами числа писать не нужно.
- На рисунке схема дорог Н-ского района изображена в виде графа, в таблице содержатся сведения о протяжённости каждой из этих дорог (в километрах). Так как таблицу и схему рисовали независимо друг от

друга, то нумерация населённых пунктов в таблице никак не связана с буквенными обозначениями на графе. Определите, какова длина дороги из пункта В в пункт С. В ответе запишите целое число.

	п1	п2	п3	п4	п5	п6	п7
п1		19		17		15	
п2	19		1		9		
п3		1			11		12
п4	17				17	25	5
п5		9	11	17			10
п6	15			25			
п7			12	5	10		



3. Функция F задаётся выражением $(\neg w \vee (((x \wedge y) \rightarrow \neg z) \wedge (x \vee y \vee z))) \wedge ((z \wedge w) \rightarrow x)$. На рисунке приведён частично заполненный фрагмент таблицы истинности функции F , содержащий неповторяющиеся строки. Определите, какому столбцу таблицы истинности функции F соответствует каждая из переменных x, y, z, w .

				F
		1	0	0
	1	0		0
	0	0		0
				0

В ответе напишите буквы x, y, z, w в том порядке, в котором идут соответствующие им столбцы (сначала буква, соответствующая первому столбцу; затем буква, соответствующая второму столбцу, и т.д.). Буквы в ответе пишите подряд, никаких разделителей между буквами ставить не нужно.

Пример. Функция F задана выражением $\neg x \vee y$, зависящим от двух переменных, а фрагмент таблицы имеет следующий вид.

		F
0	1	0

В этом случае первому столбцу соответствует переменная y , а второму столбцу — переменная x . В ответе следует написать: yx .

4. Ниже представлены два фрагмента таблиц из базы данных о жителях микрорайона. Каждая строка таблицы 2 содержит информацию о ребёнке и об одном из его родителей. Информация представлена значением поля ID в соответствующей строке таблицы 1.

На основании приведённых данных определите количество людей, у которых два ребенка разного пола. При вычислении ответа учитывайте только информацию из приведённых фрагментов таблиц.

Таблица 1			
ID	Фамилия_И.О.	Пол	Год_рождения
14	Иванова М.А.	Ж	1942
24	Сканави И.П.	М	1943
25	Сканави П.И.	М	1973
26	Сканави П.П.	М	1996

Таблица 2	
ID_Родителя	ID_Ребёнка
24	25
44	25
25	26
64	26

34	Кущенко А.И.	Ж	1964	24	34
35	Кущенко В.С.	Ж	1987	44	34
36	Кущенко С.С.	М	1964	34	35
44	Лебедь А.С.	Ж	1941	36	35
45	Лебедь В.А.	М	1953	14	36
46	Гросс О.С.	Ж	1992	34	46
47	Гросс П.О.	М	2009	36	46
54	Клеменко А.П.	Ж	1993	25	54
64	Крот П.А.	Ж	1964	64	54
68	Сергеев А.П.	М	1958	46	47
...	...	...	...	...	...

5. По каналу связи передаются сообщения, содержащие только буквы из набора: Е, С, О, Л, А, П, Р. Для передачи используется двоичный код, удовлетворяющий прямому условию Фано, согласно которому никакое кодовое слово не является началом другого кодового слова. Это условие обеспечивает возможность однозначной расшифровки закодированных сообщений. Кодовые слова для некоторых букв известны: С – 0, П – 111. Для пяти оставшихся букв Е, О, Л, А, Р кодовые слова неизвестны. Какое количество двоичных знаков потребуется для кодирования слова ЛЕСОПОЛОСА, если известно, что оно закодировано минимально возможным количеством двоичных знаков?

6. На вход алгоритма подаётся натуральное число $N > 100$. Алгоритм строит по нему новое число R следующим образом:

1. Строится двоичная запись числа N .
2. Подсчитывается количество нулей и единиц в полученной записи. Если их количество одинаково, то в конец записи добавляется цифра 0, если последняя цифра числа равна 1, и добавляется 1, если последняя цифра числа равна 0. Если количество нулей и единиц не одинаково, в конец записи добавляется цифра, которая встречается реже.
3. Шаг 2 повторяется ещё один раз.
4. Результат переводится в десятичную систему счисления.

При каком наименьшем исходном числе N в результате работы алгоритма получится число, которое при делении на 4 даёт остаток 3?

7. Исполнитель Черепаха действует на плоскости с декартовой системой координат. В начальный момент Черепаха находится в начале координат, её голова направлена вдоль положительного направления оси ординат, хвост опущен. При опущенном хвосте Черепаха оставляет на поле след в виде линии. В каждый конкретный момент известно положение исполнителя и направление его движения. У исполнителя существуют две команды:

Вперёд n (где n – целое число), вызывающая передвижение Черепахи на n единиц в том направлении, куда указывает её голова, и

Направо m (где m – целое число), вызывающая изменение направления движения на m градусов по часовой стрелке.

Запись **Повтори k [Команда1 Команда2 ... КомандаS]** означает, что последовательность из S команд повторится k раз.

Черепахе был дан для исполнения следующий алгоритм:

Повтори 31 [Вперёд 9 Направо 60 Направо 120 Вперёд 9 Направо 36].

Определите, из какого количества отрезков будет состоять фигура, заданная данным алгоритмом.

8. Музыкальный фрагмент был записан в формате моно, оцифрован и сохранён в виде файла без использования сжатия данных. Размер полученного файла – 112 Мбайт. Затем тот же музыкальный фрагмент

был записан повторно в формате стерео (двухканальная запись) и оцифрован с разрешением в 3,5 раза больше и частотой дискретизации в 2 раза меньше, чем в первый раз. Сжатие данных не производилось. Укажите размер полученного при повторной записи файла в Мбайт. В ответе запишите только целое число, единицу измерения писать не нужно.

9. Определите количество пятизначных чисел, не кратных 3, записанных в шестеричной системе счисления, в записи которых только одна цифра 2, при этом чётные и нечётные цифры не стоят рядом.

10. Дан фрагмент электронной таблицы. Из ячейки C2 в ячейку D3 была скопирована формула. При копировании адреса ячеек в формуле автоматически изменились. Каким стало числовое значение формулы в ячейке D3? В ответе запишите только число.

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

Примечание. Знак \$ обозначает абсолютную адресацию.

11. В терминологии сетей TCP/IP маской сети называется двоичное число, определяющее, какая часть IP-адреса узла сети относится к адресу сети, а какая – к адресу самого узла в этой сети. Обычно маска записывается по тем же правилам, что и IP-адрес, – в виде четырёх байтов, причём каждый байт записывается в виде десятичного числа. При этом в маске сначала (в старших разрядах) стоят единицы, а затем с некоторого разряда – нули. Адрес сети получается в результате применения поразрядной конъюнкции к заданному IP-адресу узла и маске. Например, если IP-адрес узла равен 231.32.255.131, а маска равна 255.255.240.0, то адрес сети равен 231.32.240.0.

Для узла с IP-адресом 55.102.221.201 адрес сети равен 55.102.221.192. Чему наибольшее значение последнего (крайнего справа) байта маски? Ответ запишите в виде десятичного числа.

12. При регистрации в компьютерной системе каждому пользователю присваивается идентификатор фиксированной длины, состоящий из двух частей. Первая часть включает 19 заглавных и прописных латинских букв; каждый символ кодируется отдельно с использованием минимально возможного количества битов. Вторая часть – целое число от 0001 до 3000, для его кодирования используется минимальное число бит. Для кодирования полного идентификатора выделяется целое число байтов. Кроме того, для каждого пользователя хранятся дополнительные сведения (также целое число байтов, одинаковое для каждого пользователя). Определите, сколько байтов занимают дополнительные сведения для одного пользователя, если данные о 72 пользователях занимают 4536 байтов. *Примечание:* в латинском алфавите 26 букв.

13. Исполнитель Редактор получает на вход строку цифр и преобразовывает её. Редактор может выполнять две команды, в обеих командах *v* и *w* обозначают цепочки цифр.

А) **заменить** (*v, w*).

Эта команда заменяет в строке первое слева вхождение цепочки *v* на цепочку *w*. Например, выполнение команды **заменить** (111, 27) преобразует строку 05111150 в строку 0527150. Если в строке нет вхождений цепочки *v*, то выполнение команды **заменить** (*v, w*) не меняет эту строку.

Б) **нашлось** (*v*).

Эта команда проверяет, встречается ли цепочка *v* в строке исполнителя Редактор. Если она встречается, то команда возвращает логическое значение «истина», в противном случае возвращает значение «ложь». Строка исполнителя при этом не изменяется.

Цикл

ПОКА *условие*

последовательность команд

КОНЕЦ ПОКА

выполняется, пока условие истинно.

В конструкции

ЕСЛИ условие

ТО команда1

ИНАЧЕ команда2

КОНЕЦ ЕСЛИ

выполняется *команда1* (если *условие* истинно) или *команда2* (если *условие* ложно).

Известно, что начальная строка начинается с цифры 0 и содержит 25 цифр 1, 30 цифр 2 и n цифр 3, расположенных в произвольном порядке. При каком наименьшем значении n сумма цифр в строке, получившейся в результате работы алгоритма, будет кратна 100?

НАЧАЛО

ПОКА нашлось (01) ИЛИ нашлось (02) ИЛИ нашлось (03)

 заменить (01, 1103)

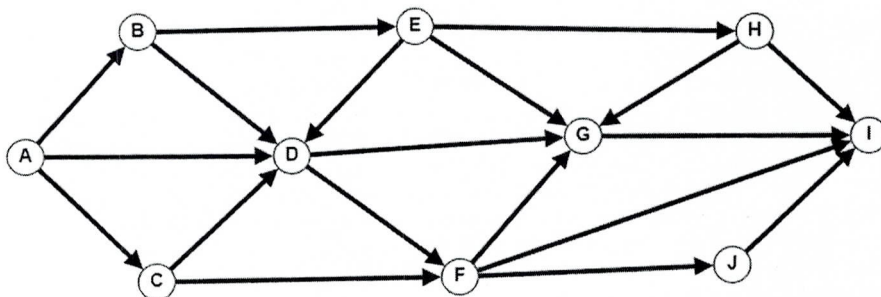
 заменить (02, 201)

 заменить (03, 20)

КОНЕЦ ПОКА

КОНЕЦ

14. На рисунке представлена схема дорог, связывающих города A, B, C, D, E, F, G, H, I, J. По каждой дороге можно двигаться только в одном направлении, указанном стрелкой. Сколько существует различных путей из города A в город I, проходящих через город G?



15. Значение арифметического выражения

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

записали в системе счисления с основанием 7. Сколько цифр 6 содержится в этой записи?

16. На числовой прямой даны три отрезка: $A=[5; 42]$, $B=[77; 120]$ и $C=[21; 97]$. Укажите количество целых значений x , для которых логическое выражение

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

ложно (т.е. принимает значение 0). Логическое утверждение $\text{ДЕЛ}(n, m)$ означает «натуральное число n делится без остатка на натуральное число m ».

17. Ниже на трех языках записан рекурсивный алгоритм F. Запишите подряд без пробелов и разделителей все числа, которые будут выведены на экран при выполнении вызова $F(6)$. Числа должны быть записаны в том же порядке, в котором они выводятся на экран.

Паскаль	Python	C++
procedure F(n: integer);	def F(n):	void F(int n) {

<pre>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>if n > 0: if (n % 2 != 0): F(n + 1) print(n + 3) else: print(n + 2) F(n - 3)</pre>	<pre>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. В языке запросов поискового сервера для обозначения логической операции «ИЛИ» используется символ «|», а для обозначения логической операции «И» – символ «&». В таблице приведены запросы и количество найденных по ним страниц некоторого сегмента сети Интернет.

Запрос	Найдено страниц (в сотнях тысяч)
<i>Логика & Информатика</i>	150
<i>Программирование</i>	750
<i>Логика</i>	375
<i>Информатика & Программирование</i>	230
<i>Информатика Программирование Логика</i>	1265
<i>Логика & Программирование</i>	0

Какое количество страниц (в сотнях тысяч) будет найдено по запросу *Информатика*?

19. В программе используется одномерный целочисленный массив *mas* с индексами от 0 до 9. Значения элементов равны 34, 29, 28, 3, 9, 6, 40, 7, 50, 4, соответственно, т.е. *mas*[0] = 34, *mas*[1] = 29 и т.д. Определите значение переменной *s* после выполнения следующего фрагмента программы (записанного ниже на трех языках программирования).

Паскаль	Python	C++
<pre>s := 0; n := 0; 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. Ниже на трёх языках программирования записан алгоритм. Укажите наименьшее значение *a*, большее 100, при вводе которого алгоритм напечатает 17.

Паскаль	Python	C++
<pre>var a, b, t :integer; begin readln(a);</pre>	<pre>a = int(input()) b = a - 34 while a != b:</pre>	<pre>int a, b, t; std::cin>>a; b = a - 34;</pre>

<pre> 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 -= b if b > a: t = a a = b b = t print(a) </pre>	<pre> while (a != b){ a -= b; if (b > a){ t = a; a = b; b = t; } } std::cout<<a<<std::endl; </pre>
--	---	---

21. Ниже на трёх языках программирования записан алгоритм. Напишите в ответе число, которое будет выведено в результате его выполнения.

Паскаль	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. В таблице представлена информация о совокупности N вычислительных процессов, которые могут выполняться параллельно или последовательно. Будем говорить, что процесс В зависит от процесса А, если для выполнения процесса В необходимы результаты выполнения процесса А. В этом случае процессы могут выполняться только последовательно.

В первом столбце таблицы указан идентификатор процесса (ID), во втором столбце таблицы – время его выполнения в миллисекундах, в третьем столбце перечислены с разделителем «;» ID процессов, от которых зависит данный процесс. Если процесс является независимым, то в таблице указано значение 0.

ID процесса В	Время выполнения процесса В (мс)	ID процесса(ов) А
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;
----	---	------

Определите **минимальное** время, через которое завершится выполнение всей совокупности процессов, при условии, что все независимые друг от друга процессы могут выполняться параллельно.

23. Исполнитель преобразует число на экране. У Исполнителя есть три команды, которым присвоены номера:

1. Прибавить 2
2. Прибавить 3
3. Умножить на 5

Программа для Исполнителя — это последовательность команд. Сколько существует программ, для которых при исходном числе 1 результатом является число 25, и при этом траектория вычислений содержит число 8 и не содержит число 20?

Траектория вычислений программы — это последовательность результатов выполнения всех команд программы. Например, для программы 123 при исходном числе 5 траектория будет состоять из чисел 7, 10, 50.

Часть 2

Для записи ответов на задания этой части (25–27) используйте БЛАНК ОТВЕТОВ № 2. Запишите сначала номер задания (25, 26 и т. д.), а затем полное решение. Ответы записывайте чётко и разборчиво.

24. На обработку поступает натуральное число, большее 100 и не превышающее 10^9 . Нужно написать программу, которая выводит на экран количество цифр числа, которые больше или равны предыдущей цифры в числе (если предыдущая существует). Программист написал программу неправильно. Ниже эта программа для Вашего удобства приведена на трёх языках программирования.

Паскаль	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; if d2 <= d1 then count += 1; d2 := d1; n = n div 10; end; writeln(p) end. </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 n //= 10 print(count) </pre>	<pre> #include <iostream> using namespace std; int main() { int n,d1,d2; cin >> n; d1 = n % 10; n = n / 10; count = 0; while (n / 10 > 0) { d2 = n % 10; if (d2 <= d1) count += 1; d2 = d1; n = n / 10; } cout << count << endl; return 0; } </pre>

1. Напишите, что выведет эта программа при вводе числа 123.
 2. Укажите максимальное трехзначное число, при вводе которого приведённая программа, несмотря на ошибки, выдаёт верный ответ.
 3. Найдите допущенные программистом ошибки и исправьте их. Исправление ошибки должно затрагивать только строку, в которой находится ошибка. Для каждой ошибки:
 - 1) выпишите строку, в которой сделана ошибка;
 - 2) укажите, как исправить ошибку, т.е. приведите правильный вариант строки.
- Известно, что в тексте программы можно исправить ровно две строки так, чтобы она стала работать правильно.

Достаточно указать ошибки и способ их исправления для одного языка программирования. Обратите внимание на то, что требуется найти и исправить ровно две ошибки в имеющейся программе, а не написать свою, возможно, использующую другой алгоритм решения.

25. Дан целочисленный массив из 50 элементов. Элементы массива пронумерованы числами от 0 до 49 и могут принимать натуральные значения от 1 до 10000 включительно. Опишите на одном из языков программирования алгоритм, который находит количество пар элементов массива, в которых оба числа в паре кратны 2, и ровно одно заканчивается на 6, а затем заменяет каждый элемент, не делящийся нацело на 6, на найденное значение. В качестве результата необходимо вывести изменённый массив, каждый элемент выводится с новой строки. Под парой подразумевается два рядом стоящих элемента массива.

Например, для исходного массива из шести элементов:

2
6
8
11
36
15

программа должна вывести следующий массив:

2
6
2
2
36
2

Паскаль	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># допускается также # использовать две # целочисленные переменные j и 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>

В качестве ответа Вам необходимо привести фрагмент программы, который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и используемую версию языка программирования, например Free Pascal 2.6). В этом случае Вы должны использовать те же самые исходные данные и переменные, какие были предложены в условии.

26. Два игрока, Петя и Ваня, играют в следующую игру. Перед игроками лежат две кучи камней. Игроки ходят по очереди, первый ход делает Петя. За один ход игрок может добавить в одну из куч (по своему выбору) три камня или увеличить количество камней в куче в **четыре** раза. Например, пусть в одной куче 10 камней, а в другой 5 камней; такую позицию в игре будем обозначать (10, 5). Тогда за один ход можно получить любую из четырёх позиций: (13, 5), (40, 5), (10, 8), (10, 20). Для того, чтобы делать ходы, у каждого игрока есть неограниченное количество камней.

Игра завершается в тот момент, когда суммарное количество камней в кучах становится не менее 105. Победителем считается игрок, сделавший последний ход, т.е. первым получивший такую позицию, при которой суммарно в кучах будет 105 или больше камней. В начальный момент в первой куче было 2 камня, во второй куче было S камней, $1 < S < 103$.

Будем говорить, что игрок имеет выигрышную стратегию, если он может выиграть при любых ходах противника. Описать стратегию игрока — значит описать, какой ход он должен сделать в любой ситуации, которая ему может встретиться при различной игре противника. В описание выигрышной стратегии не следует включать ходы играющего по этой стратегии игрока, не являющиеся для него безусловно выигрышными, т.е. не являющиеся выигрышными независимо от игры противника. Выполните следующие задания. Во всех случаях обосновывайте свой ответ.

1) Укажите значение S, при котором Петя не может выиграть за один ход, но при любом ходе Пети Ваня может выиграть своим первым ходом.

2) Найдите минимальное и максимальное значения S, при котором у Пети есть выигрышная стратегия,

причём одновременно выполняются два условия:

- Петя не может выиграть за один ход;
- Петя может выиграть своим вторым ходом независимо от того, как будет ходить Ваня.

Опишите выигрышную стратегию Пети.

3) Найдите значение S , при котором одновременно выполняются два условия:

- у Вани есть выигрышная стратегия, позволяющая ему выиграть первым или вторым ходом при любой игре Пети;
- у Вани нет стратегии, которая позволит ему гарантированно выиграть первым ходом.

Если найдено несколько значений S , в ответе запишите максимальное из них. Для указанного значения S опишите выигрышную стратегию Вани. Постройте дерево всех партий, возможных при этой выигрышной стратегии Вани (в виде рисунка или таблицы). На рёбрах дерева указывайте ходы, в узлах указывайте позиции.

27. Дана последовательность из N натуральных чисел. Числа в последовательности могут принимать значения от 1 до 10000. Рассматриваются все её непрерывные подпоследовательности такие, что сумма элементов каждой из них кратна 13, а произведение делится на 6. Найдите подпоследовательность с максимальной суммой и ее длину. Если таких подпоследовательностей найдено несколько, в ответе укажите количество элементов в самой длинной из них.

Описание входных и выходных данных

В первой строке входного файла задано количество чисел N ($1 < N < 10000000$). Каждая из следующих N строк содержит одно натуральное число в диапазоне от 1 до 10000. В качестве результата программа должна напечатать два числа: максимальную сумму непрерывной подпоследовательности, и ее длину.

Пример организации исходных данных во входном файле:

```
10
1
97
68
11
46
96
7
29
41
12
```

Пример входных данных для приведённого выше примера входных данных:

```
325 6
```

Пояснение: Сумма, равная 325, получается в результате суммирования элементов 97, 68, 11, 46, 96, 7. Среди данных элементов есть число, кратное 6, это 96.

Требуется написать эффективную по времени программу для решения описанной задачи. Программа считается эффективной по времени, если при увеличении количества элементов последовательности N в k раз время работы программы увеличивается не более чем в k раз. Максимальная оценка за правильную (не содержащую синтаксических ошибок и дающую правильный ответ при любых допустимых входных данных) программу, эффективную по времени, – 4 балла. Максимальная оценка за правильную программу, не удовлетворяющую требованиям эффективности, – 2 балла.

Вы можете сдать одну программу или две программы решения задачи (например, одна из программ может быть менее эффективна). Если Вы сдадите две программы, то каждая из них будет оцениваться независимо от другой, итоговой станет большая из двух оценок. Перед текстом программы обязательно кратко опишите алгоритм решения. Укажите использованный язык программирования и его версию.

Проверьте, чтобы каждый ответ был записан рядом с номером соответствующего задания.
--

Раздел 4 «Перечень литературы и информационных источников для подготовки к вступительным испытаниям»

- Информационный образовательный портал. Документы, материалы, пособия, пробники к ЕГЭ, ГИА. [Электронный ресурс]. – Режим доступа: <http://egeigia.ru/>;
- Официальный сайт Рособрнадзора «ЕГЭ-портал. Мы знаем о ЕГЭ все». [Электронный ресурс]. – Режим доступа: <http://4ege.ru>;
- Методические материалы и программное обеспечение по информатике - Режим доступа: <http://kpolyakov.spb.ru>
- Угринович Н. Д. Информатика и ИКТ. Профильный уровень: учебник для 10 класса – БИНОМ. Лаборатория знаний, 2012
- Угринович Н. Д. Информатика и ИКТ. Профильный уровень: учебник для 11 класса – БИНОМ. Лаборатория знаний, 2012
- В. Г. Давыдов. Программирование и основы алгоритмизации – М.: Высшая школа, 2003